

# Installation

Version: 2024.0-SNAPSHOT.33 20.01.2025 9:39:09

# Contents

---

- Installation
  - [Assumptions and versions](#)
  - [Setting up a new installation](#)
  - [Setting the license number](#)
- Configuration
  - [Basic configuration](#)
  - [External configuration](#)
  - [Spam protection](#)
- Migration
  - [Migration between environments](#)
- MultiWeb
  - [Home](#)
  - [Configuration](#)

# 1. Assumptions and versions

WebJET 2024 requires `Java 17` a `Tomcat 9` .

The basic project in gradle format can be found at [github webjetcms/basecms](https://github.com/webjetcms/basecms) (<https://github.com/webjetcms/basecms>).

In gradle projects, just specify the version in build.gradle:

```
ext {
    webjetVersion = "2024.18";
}
```

gradle

Currently there are the following versions of WebJET:

- `2024.0.47` - Stabilized version 2024.0.47 with bug fixes against version 2024.0 (without adding enhancements from SNAPSHOT version).
- `2024.40` - stabilized version 2024.40, no daily changes added
- `2024.0-SNAPSHOT` - regularly updated version from the master repository version 2024.0 compiled with Java version 17.
- `2024.18` - Stabilized version 2024.18, no daily changes added
- `2024.0.34` - Stabilized version 2024.0.34 with bug fixes against version 2024.0 (without adding enhancements from SNAPSHOT version).
- `2024.0.21` - Stabilized version 2024.0.21 with bug fixes against version 2024.0 (without adding enhancements from SNAPSHOT version).
- `2024.0.17` - Stabilized version 2024.0.17 with bug fixes against version 2024.0 (without adding enhancements from SNAPSHOT version).
- `2024.0.9` - Stabilized version 2024.0.9 with bug fixes against version 2024.0 (without adding enhancements from SNAPSHOT version).
- `2024.0` - Stabilized version 2024.0 (technically identical to 2023.52-java17), no daily changes added, compiled with Java version 17.
- `2023.52-java17` - Stabilized version 2023.52, no daily changes added, compiled with Java version 17.
- `2023.52` - Stabilized version 2023.52, no daily changes added
- `2023.40-SNAPSHOT-java17` - regularly updated version from the master repository version 2023.40 compiled with Java version 17.
- `2023.40-SNAPSHOT` - regularly updated version from the master repository version 2023.40
- `2023.40` - stabilized version 2023.40, no daily changes added
- `2023.18-SNAPSHOT-java17` - regularly updated version from the master repository version 2023.18 compiled with Java version 17.
- `2023.18-SNAPSHOT` - regularly updated version from the master repository version 2023.18
- `2023.18` - Stabilized version 2023.18, no daily changes added
- `2023.0-SNAPSHOT` - regularly updated version from the master repository version 2023.0, due to API changes this version ends before the release of version 2023.18 to avoid unexpected API changes in projects.
- `2023.0` - Stabilized version 2023.0, no daily changes added
- `2022.0-SNAPSHOT` - regularly updated version from the master repository
- `2022.52` - stabilized version 2022.52, no daily changes added
- `2022.40` - stabilized version 2022.40, no daily changes added
- `2022.18` - stabilized version 2022.18, no daily changes added
- `2022.0` - stabilized version 2022.0, no daily changes added
- `2021.0-SNAPSHOT` - regularly updated version from the master repository
- `2021.52` - stabilized version 2021.52, no daily changes added
- `2021.40` - stabilized version 2021.40, no daily changes added
- `2021.13` - Stabilized version 2021.13, no daily changes added

For version numbers:

- `YEAR.0.x` - Corrected version, **no new properties are added to it**, during the year it corrects the bugs found in WebJET CMS. The libraries used are updated only within `minor` Versions. If a library patch requires a change in `major` version cannot be incorporated into this version as it may carry the risk of changes in `API` .
- `YEAR.0-SNAPSHOT` - development version which **includes new features** and bug fixes from version `YEAR.0.x` .
- `YEAR.WEEK` - **stabilized version** from that week which is created from `SNAPSHOT` versions after successful multiple testing. Fixes for other bugs will be incorporated into the next version, no patch will be created `YEAR.WEEK.x` but the new `YEAR.WEEK` with a new number.

In case of a bug in such a version, it is therefore necessary to count on moving to the next stable version `YEAR.WEEK` or to `YEAR.0-SNAPSHOT` version.

View Full Version `YEAR.0.x` is therefore not fundamentally changed, it contains corrections of errors (unless the correction requires a fundamental change). It is suitable for use by a customer who wants to have a stable version of WebJET without adding new features during the year.

But at the same time, the version does not have to be `YEAR.0.x` the safest. If the library used in WebJET needs to be updated and it contains major changes, we cannot make this change in `YEAR.0.x` version, as it would break compatibility.

It is therefore true that `YEAR.0.x` is **most stable** in terms of changes and `YEAR.0-SNAPSHOT` is **safest** in terms of vulnerabilities.

## 1.1. Changes in the transition to 2024.0-SNAPSHOT

Similarly for [Maven Central \(https://mvnrepository.com/artifact/com.webjetcms/webjetcms\)](https://mvnrepository.com/artifact/com.webjetcms/webjetcms) version needs to be added to the `dependencies` block part `implementation("sk.iway:webjet:${webjetVersion}:libs")` :

```
dependencies {
    implementation("sk.iway:webjet:${webjetVersion}")
    implementation("sk.iway:webjet:${webjetVersion}:admin")
    implementation("sk.iway:webjet:${webjetVersion}:components")
    implementation("sk.iway:webjet:${webjetVersion}:libs")
}
```

gradle

The following libraries that are not used in the standard installation have been deleted, if your project needs them, please add them to your `build.gradle` file:

```
dependencies {
    implementation("com.amazonaws:aws-java-sdk-core:1.12.+")
    implementation("com.amazonaws:aws-java-sdk-ses:1.12.+")
    implementation("bsf:bsf:2.4.0")
    implementation("commons-validator:commons-validator:1.3.1")
    implementation("taglibs:datetime:1.0.1")
    implementation("net.htmlparser.jericho:jericho-html:3.1")
    implementation("joda-time:joda-time:2.10.13")
    implementation("io.bit3.jsass:5.1.1")
    implementation("org.jsoup:jsoup:1.15.3")
    implementation("org.mcavallo:opencloud:0.3")
    implementation("org.springframework:spring-messaging:${springVersion}")
    implementation("net.sf.uadetector:uadetector-core:0.9.22")
    implementation("net.sf.uadetector:uadetector-resources:2014.10")
    implementation("cryptix:cryptix:3.2.0")
    implementation("org.springframework:spring-messaging:${springVersion}")
    implementation("com.google.protobuf:protobuf-java:3.21.7")
    implementation("com.google.code.findbugs:jsr305:3.0.2")
    implementation("org.apache.taglibs:taglibs-standard-spec:1.2.5")
    implementation("org.apache.taglibs:taglibs-standard-impl:1.2.5")
    implementation('com.mchange:c3p0:0.9.5.5')
    implementation("xerces:xercesImpl:2.12.2")
    implementation 'jakarta.xml.bind:jakarta.xml.bind-api:2.3.3'
    implementation 'com.sun.xml.bind:jaxb-ri:2.3.3'
}
```

gradle

In the section `configurations` delete the expressions:

gradle

```
all*.exclude group: 'xml-apis', module: 'xml-apis'
all*.exclude group: 'javax.xml.stream', module: 'stax-api'
```

## 1.2. Changes when moving to GitHub/Maven Central version

- V [Maven Central \(https://mvnrepository.com/artifact/com.webjetcms/webjetcms\)](https://mvnrepository.com/artifact/com.webjetcms/webjetcms) the name of the packages is changed from `sk.iway` at `com.webjetcms` and a section is added `libs` which combines all the original `sk.iway` type dependencies `struts,daisydiff,jtidy`. V `build.gradle` edit:

gradle

```
repositories {
    mavenCentral()
    maven {
        url "https://pd4ml.tech/maven2/"
    }
    flatDir {
        dirs 'libs'
    }
}

ext {
    webjetVersion = "2024.0.3";
}

dependencies {
    implementation("com.webjetcms:webjetcms:${webjetVersion}")
    implementation("com.webjetcms:webjetcms:${webjetVersion}:admin")
    implementation("com.webjetcms:webjetcms:${webjetVersion}:components")
    implementation("com.webjetcms:webjetcms:${webjetVersion}:libs")
    ...
}
```

## 1.3. Changes in the transition to 2023.18-SNAPSHOT/2023.40

- In your project, delete the file `src/main/webapp/WEB-INF/struts-config.xml` to use the current WebJET file (from the jar file).

## 1.4. Changes when moving to Java 17

There are a few changes that need to be made to your project when moving to Java version 17. Project [basecms \(https://github.com/webjetcms/basecms/tree/release/webjet-2023-18-java17\)](https://github.com/webjetcms/basecms/tree/release/webjet-2023-18-java17) has prepared `branch`, `release/webjet-2023-18-java17` with a sample finish. V [this commit \(https://github.com/webjetcms/basecms/commit/e4b9cf6f0a88fd6f0b0cc6c57b28e7a3ec924535\)](https://github.com/webjetcms/basecms/commit/e4b9cf6f0a88fd6f0b0cc6c57b28e7a3ec924535) the complete list of changes is visible.

The simplified procedure is as follows:

Update `gradle-wrapper` to version 8 (from the original version 6), we recommend upgrading to version 7 first and then to version 8 (upgrading directly to version 8 may result in an error):

sh

```
gradlew.bat wrapper --gradle-version 7.4.2
gradlew.bat wrapper --gradle-version 8.1.1
```

After updating the gradle, you will not be able to compile the project, you also need to edit the file `build.gradle`, in which a number of `plugins`. An important change is the transition of WebJET CMS in version `java17` to the standard version `eclipselink` with the WebJET primary key generator setup.

```

plugins {
    ...
    //aktualizacia verzii pluginov
    id 'org.gretty' version "3.1.1"
    id "io.freefair.lombok" version "8.0.1"
    id "org.owasp.dependencycheck" version "8.2.1"
    ...
}

ext {
    //aktualizovana verzia WebJETu
    webjetVersion = "2023.18-SNAPSHOT-java17";
}

dependencies {
    //celu sekciu modules ZMAZAT
    modules {
        module("org.apache.struts:struts-core") { replacedBy("sk.iway:webjet", "mame integrovane") }
        module("org.apache.struts:struts-taglib") { replacedBy("sk.iway:webjet", "mame integrovane") }
        module("org.eclipse.persistence:org.eclipse.persistence.moxy") { replacedBy("sk.iway:webjet", "mame integrovane") }
        module("org.eclipse.persistence:org.eclipse.persistence.core") { replacedBy("sk.iway:webjet", "mame integrovane") }
        module("org.eclipse.persistence:org.eclipse.persistence.asm") { replacedBy("sk.iway:webjet", "mame integrovane") }
        module("org.eclipse.persistence:org.eclipse.persistence.sdo") { replacedBy("sk.iway:webjet", "mame integrovane") }
    }

    //zmazat pri prechode na verziu 2023.18+
    implementation("sk.iway:webjet:${webjetVersion}:struts")
    implementation("sk.iway:webjet:${webjetVersion}:daisydiff")
    implementation("sk.iway:webjet:${webjetVersion}:jtidy")
}

java {
    //nastavenie verzie Java na 17
    sourceCompatibility = JavaVersion.VERSION_17
    targetCompatibility = JavaVersion.VERSION_17
}

lombok {
    //nastavenie verzie lombok
    version = "1.18.28"
}

gretty {
    //zakomentovat/zmazat managedClassReload = true - nie je mozne pouzit s Java > 8
    //managedClassReload = true
}

```

groovy

Then we recommend to restart your development environment, in case of VS Code perform the action

**Java: Clean Java Language Server Workspace** to completely delete temporary files.

If you are running multiple installations of WebJET on a single Tomcat server, it is possible that older versions may not be fully compatible with Java 17:

```
[ERROR] ContextLoader - Context initialization failed <java.lang.IllegalStateException: Cannot load
configuration class: sk.iway.iwcm.system.spring.SpringSecurityConf>java.lang.IllegalStateException: Cannot load configuration class: sk.iway.iwcm.system.spring.SpringSecurityConf
...
Caused by: java.lang.reflect.InaccessibleObjectException: Unable to make protected final java.lang.Class
java.lang.ClassLoader.defineClass(java.lang.String,byte[],int,int,java.security.ProtectionDomain) throws
java.lang.ClassFormatError accessible: module java.base does not "opens java.lang" to unnamed module @4c6e3350
```

set the following for Tomcat **JAVA\_OPTS** :

```
JAVA_OPTS="$JAVA_OPTS --add-exports=java.naming/com.sun.jndi.ldap=ALL-UNNAMED --add-
opens=java.base/java.lang=ALL-UNNAMED --add-opens=java.base/java.lang.invoke=ALL-UNNAMED --add-
opens=java.base/java.io=ALL-UNNAMED --add-opens=java.base/java.security=ALL-UNNAMED --add-
opens=java.base/java.util=ALL-UNNAMED --add-opens=java.management/javax.management=ALL-UNNAMED --add-
opens=java.naming/javax.naming=ALL-UNNAMED"
```

### 1.4.1. Update from Java version 8

If you were still running Tomcat with Java version 8 you may have problems with missing libraries (these are also needed for Java 11). If you get an error in the log **java.lang.NoClassDefFoundError: javax/activation/DataSource** :

```
java.util.concurrent.ExecutionException: org.apache.catalina.LifecycleException: Failed to start component
[StandardEngine[Catalina].StandardHost[...].StandardContext[]]
...
Caused by: java.lang.NoClassDefFoundError: javax/activation/DataSource
```

it is necessary to add to each installation of WebJET CMS to the folder **WEB-INF/lib** copied libraries from [of this ZIP archive](https://docs.webjetcms.sk/latest/en/install/lib-java11.zip) (<https://docs.webjetcms.sk/latest/en/install/lib-java11.zip>) and delete files (if any):

```
jaxb-api-2.1.jar
jaxb-runtime-3.0.0-M2.jar
```

If you have been using the WebJET version **8.0-8.6** - older than **08/2019** , or you get the following error on startup:

```
[10.09 13:48:16 {vubintra} {JpaTools}] JPA: adding class: sk.iway.spirit.model.Media
[10.09 13:48:16 {vubintra} {JpaTools}] JPA: adding class: sk.iway.iwcm.io.FileHistoryBean
[10.09 13:48:16 {vubintra} {WebJETJavaSECMPIInitializer}] initPersistenceUnits[iwcm], beans=82
[10.09 13:48:16 {vubintra}] [FAIL]
java.lang.NullPointerException
    at org.eclipse.persistence.internal.jpa.EntityManagerSetupImpl.predeploy(EntityManagerSetupImpl.java:2027)
    at org.eclipse.persistence.internal.jpa.deployment.JPAInitializer.callPredeploy(JPAInitializer.java:100)
    at
sk.iway.iwcm.system.jpa.WebJETJavaSECMPIInitializer.initPersistenceUnits(WebJETJavaSECMPIInitializer.java:307)
    at sk.iway.iwcm.system.jpa.WebJETJavaSECMPIInitializer.initialize(WebJETJavaSECMPIInitializer.java:134)
```

```

at
sk.iway.iwcm.system.jpa.WebJETJavaSECMPIInitializer.getJavaSECMPIInitializer(WebJETJavaSECMPIInitializer.java:95)
at
sk.iway.iwcm.system.jpa.WebJETJavaSECMPIInitializer.getJavaSECMPIInitializer(WebJETJavaSECMPIInitializer.java:61)
at
sk.iway.iwcm.system.jpa.WebJETPersistenceProvider.createEntityManagerFactory(WebJETPersistenceProvider.java:32)

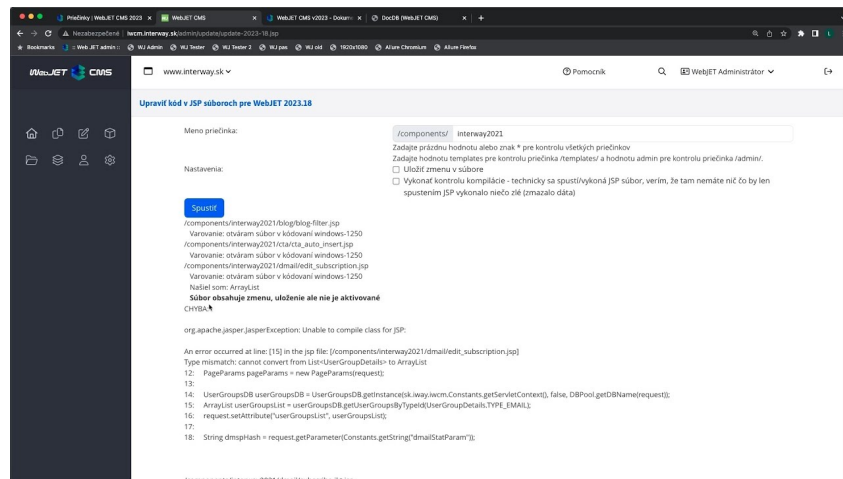
```

the library needs to be updated `eclipseLink` and the Java initialization class, download the update archive [jpa-wj82.zip](https://docs.webjetcms.sk/latest/en/install/jpa-wj82.zip) (<https://docs.webjetcms.sk/latest/en/install/jpa-wj82.zip>) and unzip it in the root folder of the web application. Overwrite the file `eclipseLink.jar` a `WebJETJavaSECMPIInitializer.class`.

## 1.5. Changes on update to 2023.18

View Full Version `2023.18` changes the API and the way distribution archives are generated. The main changes to the API are in the use of generic objects of type `List/Map` instead of specific implementations `ArrayList/Hashtable`. Therefore, you need to recompile your classes and modify the JSP files.

YouTube video: [https://www.youtube.com/watch?v=sfu5b\\_S7Q8Q](https://www.youtube.com/watch?v=sfu5b_S7Q8Q)



V `build.gradle` it is necessary to delete the part:

```

implementation("sk.iway.webjet:${webjetVersion}:struts")
implementation("sk.iway.webjet:${webjetVersion}:daisydiff")
implementation("sk.iway.webjet:${webjetVersion}:jtidy")

```

Also check the file `src/main/resources/logback.xml` in which the date and time format in `ConsoleAppender` (deleted `,SSS`, if you need to log even hundredths of a second use `.SSS`):

```
<pattern>[%X{installName}] [%c{1}] [%p] [%X{userId}] %d{yyyy-MM-dd HH:mm:ss} - %msg%n</pattern>
```

xml

To simplify the update, you can use the script `/admin/update/update-2023-18.jsp` for checking and repairing JSP files. Customer Java classes need to be recompiled and bugs need to be fixed due to API changes.

Several Java classes and packages and the corresponding JSP file are cleaned/deleted. To support the deleted parts in projects, you need to either use the appropriate WebJET NET product or bring them into the project from the version 8 source code.

More information is in [the list of changes](https://docs.webjetcms.sk/latest/en/CHANGELOG-2023) (<https://docs.webjetcms.sk/latest/en/CHANGELOG-2023>).



## 1.6. Changes compared to version 8.8

V `build.gradle` it is necessary to delete expressions compared to version 8.8:

```
compile("sk.iway:webjet:${webjetVersion}:swagger-ui")

compile 'taglibs:standard:1.1.2'
compile 'javax.servlet:jstl:1.2'

providedCompile 'org.slf4j:slf4j-log4j12:1.7.25'

exclude group: 'org.slf4j', module: 'log4j-over-slf4j'
```

gradle

and in the section `configurations` modify the exceptions as follows:

```
configurations {
    all*.exclude group: 'org.slf4j', module: 'slf4j-log4j12'
    all*.exclude group: 'org.slf4j', module: 'jcl104-over-slf4j' //je nahradene novsim jcl-over-slf4j
    all*.exclude group: 'commons-logging', module: 'commons-logging'
    all*.exclude group: 'log4j', module: 'log4j'

    //javax.xml.stream:stax-api:1.0-2 -> stax:stax-api:1.0.1
    all*.exclude group: 'javax.xml.stream', module: 'stax-api'

    grettyRunnerTomcat85 {
    }
    grettyRunnerTomcat9 {
        // gretty pouziva staru verziu commons-io, ktora koliduje s nasou
        // https://mvnrepository.com/artifact/commons-io/commons-io
        exclude group: 'commons-io', module: 'commons-io'
    }
}
```

gradle

Due to the transition from `log4j` at `logback` delete file `src/main/resources/log4j.properties` and add the file `src/main/resources/logback.xml`:

```
<configuration>
  <appender name="STDOUT" class="ch.qos.logback.core.ConsoleAppender">
    <encoder>
      <pattern>[%X{installName}] [%c{1}] [%p] [%X{userId}] %d{yyyy-MM-dd HH:mm:ss,SSS} - %msg%n</pattern>
    </encoder>
  </appender>

  <appender name="IN_MEMORY" class="sk.iway.iwcm.system.logging.InMemoryLoggerAppender" />

  <root level="ERROR">
    <appender-ref ref="STDOUT" />
    <appender-ref ref="IN_MEMORY" />
  </root>

  <logger level="INFO" name="sk.iway"/>
```

xml

```

<!--
!!! Nastavovat priamo tu len ak je to nevyhnutne potrebne !!!
Odporucame nastavit logovacie levely v localconf.jsp cez konstantu:
Constants.setString("logLevels", "sk.iway=DEBUG,org.springframework=DEBUG");

# Spring logging
<logger level="DEBUG" name="org.springframework"/>

# SQL logging
<logger level="TRACE" name="org.eclipse.persistence"/>
<logger level="DEBUG" name="org.hibernate"/>
<logger level="TRACE" name="org.hibernate.type.descriptor.sql.BasicBinder"/>

# Hibernate logging options (INFO only shows startup messages)
<logger level="INFO" name="org.hibernate"/>

# Log JDBC bind parameter runtime arguments
<logger level="TRACE" name="org.hibernate.type"/>
-->
</configuration>

```

## 1.7. Editing the generated archive

If you need to adjust for deployment `WAR` archive, you can use the following tips:

### Other web.xml file

If you need to deploy a different `web.xml` file as you use during development you can take advantage of the options offered by [gradle war](https://docs.gradle.org/current/userguide/war_plugin.html) ([https://docs.gradle.org/current/userguide/war\\_plugin.html](https://docs.gradle.org/current/userguide/war_plugin.html)) role in `build.gradle`:

```

war {
    ....
    webXml = file('src/someWeb.xml') // copies a file to WEB-INF/web.xml
}

```

### Modified logging

WebJET writes log files using [slf4j/logback](https://logback.qos.ch/manual/configuration.html) (<https://logback.qos.ch/manual/configuration.html>). This allows you to search for an additional configuration file, e.g. `logback-test.xml` to be used primarily (if any). In it you can have logging set up for development. In the file `logback.xml` you will be set for deployment. When creating `WAR` archive additional `logback-test.xml` v `build.gradle` you skip:

```

war {
    ....
    rootSpec.exclude('*/logback-*.xml')
}

```

## 1.8. Additional project setup options

### 1.8.1. Setting up social login

If you use social login (e.g. Facebook) in your project, you need to add a library to the project gradle `socialauth` . It is not included in the distribution by default because it is rarely used and also contains a potential vulnerability. You add the library in a file `build.gradle` :

```
// https://mvnrepository.com/artifact/org.brickred/socialauth
implementation group: 'org.brickred', name: 'socialauth', version: '4.15'
```

gradle

### 1.8.2. Separate web.xml file for deployment

If you need a different web.xml file for version deployment to environments than for standard development, you can use the settings options [tasks of war \(https://docs.gradle.org/current/userguide/war\\_plugin.html\)](https://docs.gradle.org/current/userguide/war_plugin.html) v `build.gradle` where it is possible to attach a different `web.xml` File:

```
war {
    zip64 = true
    webXml = file('src/web-azure.xml')
}
```

## 1.9. Changes to the database schema

New columns are added to several tables when version 2021 is enabled:

- `_properties_` - column added `update_date` , column `id` set as `autoincrement`
- `crontab` - added `task_name` column
- `_conf_prepared_` - set column `date_prepared` for the possibility to insert `NULL` Value.

### 1.9.1. MariaDB - utf8mb4 and InnoDB support

WebJET is used by default `storage engine InnoDB` , this is set in the configuration variable `mariaDbDefaultEngine` , which in WebJET 8 has the value `MyISAM` and in the new WebJET value `InnoDB` . When using `InnoDB` it is possible to use the charset `utf8mb4` with full support `utf8` coding (emoticons). By default `MyISAM` encoding is used in the tables `utf8` , which in MySQL/MariaDB is only 3 bytes and thus does not support all characters (different emoticons).

You can use the following SQL statements to create a new database schema and user:

```
CREATE DATABASE xxx_web DEFAULT CHARACTER SET utf8mb4 DEFAULT COLLATE utf8mb4_general_ci;
CREATE USER xxx_web IDENTIFIED BY 'gJ0gzNJS MwIv4Fg';
GRANT ALL PRIVILEGES ON xxx_web.* TO `xxx_web`@`%`;
FLUSH PRIVILEGES;
```

sql

## 1.10. Rollback of changes

Changes to the database schema are backwards compatible with version 8.8, by default there is no need to revert schema changes. However, if you need to revert the changes, use the following SQL statements:

MySQL / Microsoft SQL:

sql

```
ALTER TABLE _properties_ DROP COLUMN update_date;
ALTER TABLE _properties_ DROP COLUMN id;
DELETE FROM _db_ WHERE note='13.5.2020 [pgajdos] Pridanie stlpca update_date do _properties_ tabulky';

ALTER TABLE crontab DROP COLUMN task_name;
DELETE FROM _db_ WHERE note='08.07.2020 [pgajdos] Pridanie stlpca task_name do tabulky crontab, umiestnenie
stlpca na druhe miesto';
DELETE FROM _db_ WHERE note='09.07.2020 [pgajdos] Zapisanie popisov cronjobov do stlpca task_name, pre tabulku
crontab';

ALTER TABLE documents DROP COLUMN temp_field_a_docid;
ALTER TABLE documents DROP COLUMN temp_field_b_docid;
ALTER TABLE documents DROP COLUMN temp_field_c_docid;
ALTER TABLE documents DROP COLUMN temp_field_d_docid;
DELETE FROM _db_ WHERE note='14.12.2021 [sivan] Pridanie stlpcov temp_field_a_docid ... temp_field_d_docid do
tabulky documents';

ALTER TABLE documents_history DROP COLUMN temp_field_a_docid;
ALTER TABLE documents_history DROP COLUMN temp_field_b_docid;
ALTER TABLE documents_history DROP COLUMN temp_field_c_docid;
ALTER TABLE documents_history DROP COLUMN temp_field_d_docid;
DELETE FROM _db_ WHERE note='14.12.2021 [sivan] Pridanie stlpcov temp_field_a_docid ... temp_field_d_docid do
tabulky documents_history';

ALTER TABLE documents DROP COLUMN show_in_navbar;
ALTER TABLE documents DROP COLUMN show_in_sitemap;
ALTER TABLE documents DROP COLUMN logged_show_in_menu;
ALTER TABLE documents DROP COLUMN logged_show_in_navbar;
ALTER TABLE documents DROP COLUMN logged_show_in_sitemap;
DELETE FROM _db_ WHERE note='14.12.2021 [sivan] Pridanie boolean stlpcov (show_in_navbar, show_in_sitemap,
logged_show_in_menu, logged_show_in_navbar, logged_show_in_sitemap) do tabulky documents';

ALTER TABLE documents_history DROP COLUMN show_in_navbar;
ALTER TABLE documents_history DROP COLUMN show_in_sitemap;
ALTER TABLE documents_history DROP COLUMN logged_show_in_menu;
ALTER TABLE documents_history DROP COLUMN logged_show_in_navbar;
ALTER TABLE documents_history DROP COLUMN logged_show_in_sitemap;
DELETE FROM _db_ WHERE note='18.12.2021 [sivan] Pridanie boolean stlpcov (show_in_navbar, show_in_sitemap,
logged_show_in_menu, logged_show_in_navbar, logged_show_in_sitemap) do tabulky documents_history';

ALTER TABLE groups DROP COLUMN show_in_navbar;
ALTER TABLE groups DROP COLUMN show_in_sitemap;
ALTER TABLE groups DROP COLUMN logged_show_in_navbar;
ALTER TABLE groups DROP COLUMN logged_show_in_sitemap;
DELETE FROM _db_ WHERE note='28.12.2021 [sivan] Pridanie boolean stlpcov (show_in_navbar, show_in_sitemap,
logged_show_in_navbar, logged_show_in_sitemap) do tabulky groups.';
```

```
ALTER TABLE groups_scheduler DROP COLUMN show_in_navbar;
ALTER TABLE groups_scheduler DROP COLUMN show_in_sitemap;
ALTER TABLE groups_scheduler DROP COLUMN logged_show_in_navbar;
ALTER TABLE groups_scheduler DROP COLUMN logged_show_in_sitemap;
DELETE FROM _db_ WHERE note='28.12.2021 [sivan] Pridanie boolean stlpcov (show_in_navbar, show_in_sitemap,
logged_show_in_navbar, logged_show_in_sitemap) do tabulky groups_scheduler.';
```

Oracle:

```
sql
DROP TRIGGER T_webjet_properties;
DROP SEQUENCE S_webjet_properties;
ALTER TABLE webjet_properties DROP COLUMN update_date;
ALTER TABLE webjet_properties DROP COLUMN id;
DELETE FROM webjet_db WHERE note='13.5.2020 [pgajdos] Pridanie stlpca update_date do _properties_ tabulky';

ALTER TABLE crontab DROP COLUMN task_name;
DELETE FROM webjet_db WHERE note='08.07.2020 [pgajdos] Pridanie stlpca task_name do tabulky crontab,
umiestnenie stlpca na druhe miesto';
DELETE FROM webjet_db WHERE note='09.07.2020 [pgajdos] Zapisanie popisov cronjobov do stlpca task_name, pre
tabulku crontab';
```

## 2. Setting up a new installation

Instructions for creating a new installation/clean database for a new project in WebJET. Installation is only allowed on the domain for security reasons `localhost`, after installation you can use the standard domain name.

### 2.1. Prerequisites

- locally functional WebJET (configured Tomcat, web application)
- access to a DB server with rights to create a new DB schema (or an already established DB schema)

### 2.2. Basic server requirements

- Server with at least 8GB of memory (for applications with heavier workloads, at least 12GB), processor with at least `Dual Core 2 GHz` (for servers with a heavier Quad core load), disk space of at least 40GB.
- Database `MySQL/MariaDB verzie 5.0+` (UTF-8 encoded), or the database `Microsoft SQL 2012+` or database `Oracle 11g+` or `PostgreSQL 16+`.
- [Open JDK \(https://adoptium.net/temurin/releases/\)](https://adoptium.net/temurin/releases/) version 17 and application server [Tomcat \(https://tomcat.apache.org/download-90.cgi\)](https://tomcat.apache.org/download-90.cgi) 9.
- Connection to SMTP server for sending emails.
- Functional DNS server.
- To speed up the generation of preview images, we recommend the installed library [ImageMagick \(https://imagemagick.org/script/download.php\)](https://imagemagick.org/script/download.php).

For installations of products such as `NET`, `LMS`, `DSK` the minimum requirements suitable for installations of up to 50 users (25 working simultaneously). For a higher number of users, it is necessary to increase the RAM and the CPU appropriately - for each additional 50 users working simultaneously +4GB of memory and 1CPU. For more than 200 users we recommend a cluster solution.

For installations of products such as `NET`, `LMS`, `DSK` it is necessary to enable on the server `websocket` connect and install the server [RabbitMQ \(https://www.rabbitmq.com/\)](https://www.rabbitmq.com/).

## 2.3. Creating a DB schema

- connect to the DB server and create a new database/schema (if not already established)
- in the file `src/main/resources/poolman.xml` in the gradle project, or `/WEB-INF/classes/poolman.xml` when using a ready-made WAR file, set up a database connection:

MariaDB (<https://mariadb.com/kb/en/library/about-mariadb-connector-j/>):

```
<?xml version="1.0" encoding="UTF-8"?>

<poolman><management-mode>local</management-mode>
  <datasource>
    <dbname>iwcm</dbname>
    <driver>com.mysql.jdbc.Driver</driver>
    <url>jdbc:mysql://MEN0-SQL-SERVERA/MEN0-SCHEMY</url>
    <username>DB-LOGIN</username>
    <password>DB-HESLO</password>
  </datasource>
</poolman>
```

xml

Microsoft SQL (<http://jtds.sourceforge.net/faq.html>):

```
<?xml version="1.0" encoding="UTF-8"?>

<poolman><management-mode>local</management-mode>
  <datasource>
    <dbname>iwcm</dbname>
    <driver>net.sourceforge.jtds.jdbc.Driver</driver>
    <url>jdbc:jtds:sqlserver://MEN0-SQL-SERVERA:1433/MEN0-SCHEMY;encoding=utf-8</url>
    <username>DB-LOGIN</username>
    <password>DB-HESLO</password>
  </datasource>
</poolman>
```

xml

Oracle (<https://docs.oracle.com/en/database/oracle/oracle-database/23/jajdb/>):

```
<?xml version="1.0" encoding="UTF-8"?>

<poolman><management-mode>local</management-mode>
  <datasource>
    <dbname>iwcm</dbname>
    <driver>oracle.jdbc.OracleDriver</driver>
    <url>jdbc:oracle:thin:@MEN0-SQL-SERVERA:1521/MEN0-INSTANCIE</url>
    <username>DB-LOGIN</username>
    <password>DB-HESLO</password>
  </datasource>
</poolman>
```

xml

PostgreSQL (<https://jdbc.postgresql.org/documentation/use/>):

xml

```
<?xml version="1.0" encoding="UTF-8"?>

<poolman><management-mode>local</management-mode>
  <datasource>
    <dbname>iwcm</dbname>
    <driver>org.postgresql.Driver</driver>
    <url>jdbc:postgresql://localhost/DB-NAME?currentSchema=webjet_cms</url>
    <username></username>
    <password></password>
  </datasource>
</poolman>
```

The following XML elements are supported:

- `dbname` - the name of the database connection, for WebJET CMS tables it must have the value `iwcm` , but you can set more in XML `datasource` elements and create a connection to other databases, set a unique name here
- `driver` - java database driver class
- `url` - URL in the format for `JDBC` Connection
- `username` - login name
- `password` - login password

Optionally, it is possible to set:

- `initialConnections` - default number of open database connections (default 0)
- `minimumSize` - minimum number of constantly open database connections (default 0)
- `maximumSize` - maximum number of open database connections (default 50)
- `connectionTimeout` - number of seconds when the connection is considered not closed (default 300)
- `autoCommit` - if set to true is set `connection.setAutoCommit(true);` (default `false` )
- `testQuery` - test SQL expression to verify the functionality of the connection. For `JDBC` v4 drivers are used to call `isValid()` , for older drivers need to be set. Value `true` sets the default expression `SELECT 1` (used automatically for `jtds` controller). But it is possible to set your own SQL expression.

## 2.4. Fulfillment of the DB scheme

WebJET includes a built-in configuration that can populate an empty DB schema.

- run WebJET/Tomcat
- WebJET reports an error (multiple errors) at startup

log

```
[27.11 8:32:49 {webjet} {InitServlet}] -----
[27.11 8:32:49 {webjet} {InitServlet}] WebJET initializing, root: ...
[27.11 8:32:49 {webjet} {InitServlet}]
[27.11 8:32:49 {webjet} {InitServlet}] Checking database connection:
[27.11 8:32:49 {webjet} {InitServlet}] Database connection: [OK]
java.sql.SQLException: Table 'MEN0-SCHEMY._conf_' doesn't exist
...
[27.11 8:32:49 {webjet} {InitServlet}] ERROR: Server not configured.
[27.11 8:32:49 {webjet} {InitServlet}] ERROR: Server not configured.
[27.11 8:32:49 {webjet} {InitServlet}] ERROR: Server not configured.
...
```

When you try to log in or access the WebJET website, you will receive an error message:

Nastala chyba pri komunikácii s databazou, skuste sa prosim pripojit neskor.  
Error occurred during database query, please try connecting later.

- Open the URL in your browser [installations \(http://localhost/wjerrorpages/setup/setup\)](http://localhost/wjerrorpages/setup/setup).

WebJET everything that starts with `/wjerrorpages/` will process even if it is not started. Automatically provides a static file `/wjerrorpages/dberror.html` (<http://localhost/wjerrorpages/dberror.html>) for any GET request. In the directory `/wjerrorpages/` it is also possible to have images, but we recommend to insert them via `data:` entry directly into `dberror.html`.

- The above URL has an exception and is allowed to be used even if WebJET is not started correctly (but only on the domain `localhost` or `iwcm.interway.sk`).
- You will be presented with the WebJET installation dialog:

**WebJET nastavenie**

Toto je úvodné nastavenie pre WebJET, vyplňte požadované konfiguračné údaje

**Konfigurácia databázy**

Typ databázy:

IP adresa/názov domény:

Port:  nechajte prázdne pre predvolené nastavenie

Prihlasovacie meno:

Heslo:

Parametre:

Názov databázy:

Použitie iného používateľa pre príkaz "CREATE DATABASE": ☐

**Konfigurácia systému**

Názov inštalácie:

Licenčný kľúč:

SMTP Server:

Predvolený jazyk:

Kódovanie znakov:

- Check/enter the data for setting the database connection (defaults to the values from the poolman.xml file). The installation makes the connection directly to those values (ignoring the values in poolman.xml), so it needs them. But if the file `poolman.xml` already exists, it will not overwrite it, so for the next start the values in `poolman.xml`. If the file does not exist, it will be created according to the specified values.
- Enter a unique installation name (without spaces and accents, e.g. `interway2023`) and license number (if you are not using the Open Source version) and check the other values.
- Click OK to start the installation. If the validation of the entered values is successful, you will see the following message:



**WebJET successfully configured**

WebJET is configured. Restart the application server, wait about 5 minutes, and then [login to the admin area](#).

After login, please change admin password and update WebJET to the latest version.

Thank you for using WebJET, if you have any questions or comments, please [contact us](#).

In the background WebJET fills the initial data (according to `/WEB-INF/sql/blank_web_DBTYPE.sql`) and then performs a reboot. If the reboot is not performed automatically (the server is not set to reboot automatically), restart the application server manually.

In the log you should see something like:

```
fillEmptyDatabaseMySQL
fillEmptyDatabaseMySQL 1
fillEmptyDatabaseMySQL 2
hasDatabase=false

#
# Table structure for table '_conf_'
#

CREATE TABLE _conf_ (
  name varchar(255) NOT NULL default '',
  value varchar(255) NOT NULL default '',
  UNIQUE KEY name (name)
) ENGINE=MyISAM;

...

#
# Dumping data for table 'users'
#

INSERT INTO users VALUES("1", "", "", "Administrátor", "admin", "d7ed8dc6fc9b4a8c3b442c3dcc35bfe4", "1", NULL,
"Interway s.r.o.", "Hattalova 12/a", "", "lubos.balat@interway.sk", "83103", "Slovakia", "0903-450445", "1",
"", NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL)
Executing:
INSERT INTO users VALUES("2", "", "Obchodny", "Partner", "partner", "34f414bd2609b73403ea09787fb0aac4", "0",
"2", "Interway s.r.o.", "Hattalova 12/a", "", "lubos.balat@interway.sk", "83103", "Slovensko", "0903-945990",
"1", "", NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL)
Executing:
INSERT INTO users VALUES("3", "", "VIP", "Klient", "vipklient", "d1a9b4b9977e4829011396ec9dd2cf6a", "0", "1",
"Interway s.r.o.", "Hattalova 12/a", NULL, "lubos.balat@interway.sk", "83103", "Slovensko", "0903-945990", "1",
NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL, NULL)
[27.11 9:24:31 {webjet} {InitServlet}] RESTART request ret=true
[27.11 9:24:31 {webjet} {InitServlet}] RESTART request ret=true
```

After a reboot, a schema update is performed according to `autoupdate.xml`:

```
PathFilter init
PathFilterInit - customPath: /Users/jeeff/Documents/DISK_E/webapps-server/ppa
```

```

[27.11 9:25:05 {webjet} {InitServlet}] init start
[27.11 9:25:05 {webjet} {InitServlet}] contextDbName=null
Constants - clearValues
[27.11 9:25:05 {webjet} {InitServlet}] dbName=iwcm
[27.11 9:25:05 {webjet} {InitServlet}] -----
[27.11 9:25:05 {webjet} {InitServlet}] WebJET initializing, root: /Users/jeeff/Documents/workspace-
idea/webjet8/WebContent/
[27.11 9:25:05 {webjet} {InitServlet}]
[27.11 9:25:05 {webjet} {InitServlet}] Checking database connection:
[27.11 9:25:05 {webjet} {InitServlet}] Database connection: [OK]

...

[27.11 9:25:06 {webjet}] update database call
[27.11 9:25:06 {webjet}] ----- Updating database [DBType=3] -----
[27.11 9:25:06 {webjet}] 18.5.2004 [jeeff] vo vyhľadavani statistiky sa eviduje remote host [27.11 9:25:06
{webjet}] count=1 [27.11 9:25:06 {webjet}] [1/1] [27.11 9:25:06 {webjet}] [OK] [27.11 9:25:06 {webjet}] [OK]
[27.11 9:25:06 {webjet}] 24.5.2004 [jeeff] tabuľka s tipmi dna [27.11 9:25:06 {webjet}] count=1 [27.11
9:25:06 {webjet}] [1/1] [27.11 9:25:06 {webjet}] [OK] [27.11 9:25:06 {webjet}] [OK]
[27.11 9:25:06 {webjet}] 9.6.2004 [joruz] zoznam alarmov pre notifikáciu registrácie [27.11 9:25:06
{webjet}] count=1 [27.11 9:25:06 {webjet}] [1/1] [27.11 9:25:06 {webjet}] [OK] [27.11 9:25:06 {webjet}] [OK]

...

[27.11 9:25:39 {webjet}] MeninyImport constructor
[27.11 9:25:39 {webjet}] -> loading prop [sk]: /text.properties
[27.11 9:25:41 {webjet}] ExcelImportJXL doImport: sheet=meniny
[27.11 9:25:41 {webjet}] header[0]=day;
[27.11 9:25:41 {webjet}] header[1]=month;
[27.11 9:25:41 {webjet}] header[2]=name;
[27.11 9:25:41 {webjet}] header[3]=lng;
[27.11 9:25:41 {webjet}] importujem meniny: 1.1 (Nový rok, Deň vzniku SR) [sk]
[27.11 9:25:41 {webjet}] importujem meniny: 2.1 Alexandra [sk]
[27.11 9:25:41 {webjet}] importujem meniny: 3.1 Daniela [sk]
[27.11 9:25:41 {webjet}] importujem meniny: 4.1 Drahoslav [sk]
[27.11 9:25:41 {webjet}] importujem meniny: 5.1 Andrea [sk]
[27.11 9:25:41 {webjet}] importujem meniny: 6.1 Antónia (Zjavenie pána, Traja králi) [sk]

...

[27.11 9:26:07 {webjet}] 10 tasks should run on this node
[27.11 9:26:07 {webjet}] ----- INIT DONE -----
PathFilterInit - customPath: /Users/jeeff/Documents/DISK_E/webapps-server/ppa
[27.11 9:26:07 {webjet}] ----- INIT DONE indexed -----
[27.11 9:26:07 {webjet}] runRefresh start

...

[webjet][s.i.i.s.s.BaseSpringConfig][INFO][0] 2023-09-29 12:18:13 - -----> Configure security,
http=org.springframework.security.config.annotation.web.builders.HttpSecurity@364552cf
[webjet][s.i.i.s.s.SpringSecurityConf][INFO][0] 2023-09-29 12:18:13 - configure - SpringAppInitializer - end -
sk.iway.iwcm.system.spring.BaseSpringConfig
[webjet][s.i.i.s.s.SpringSecurityConf][INFO][0] 2023-09-29 12:18:13 - configure - SpringAppInitializer - start
- sk.iway.webjet.v9.V9SpringConfig
[webjet][s.i.i.s.s.SpringSecurityConf][INFO][0] 2023-09-29 12:18:13 - configure - SpringAppInitializer - end -
sk.iway.webjet.v9.V9SpringConfig

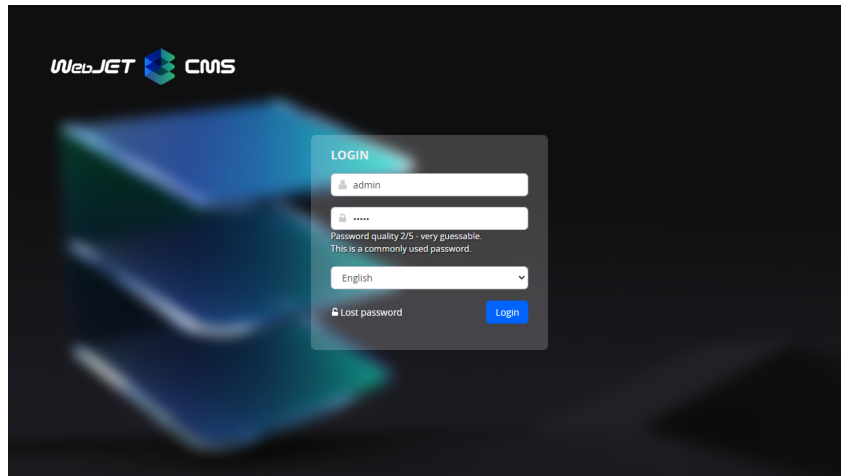
```

```
[webjet][s.i.i.s.s.SpringSecurityConf][INF0][0] 2023-09-29 12:18:13 - configure - SpringAppInitializer - start
- sk.iway.webjet_init.SpringConfig
[webjet][s.i.i.s.s.SpringSecurityConf][INF0][0] 2023-09-29 12:18:13 - configure - SpringAppInitializer - end -
sk.iway.webjet_init.SpringConfig
```

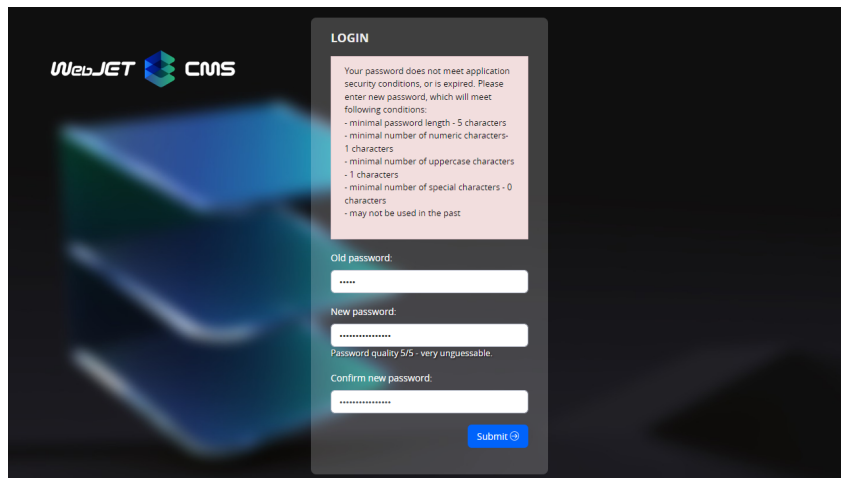
At this point, WebJET is initialized and booted to the default state.

## 2.5. Login to administration

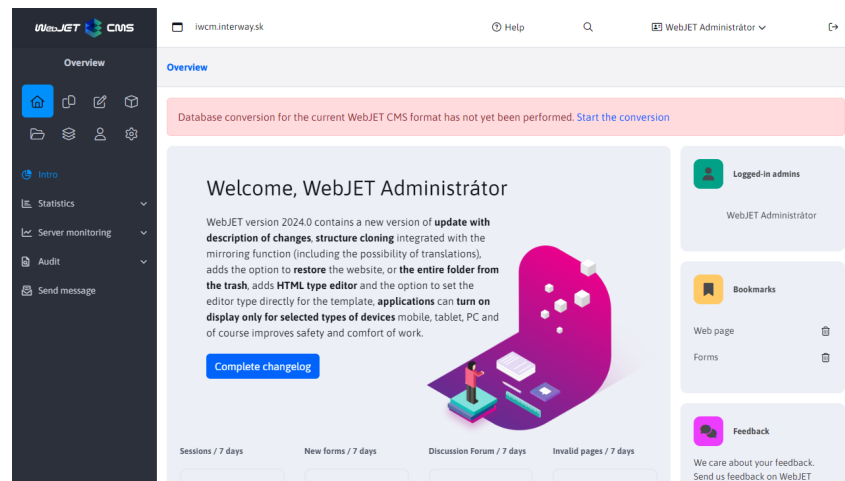
Log in to [section admin \(http://localhost/admin/\)](http://localhost/admin/) named `admin` and the password `heslo` :



WebJET will prompt you to change your password:



After logging in, you may see a message on the home screen that the database conversion has not been performed. Click on the link [Start the conversion \(http://localhost/admin/update/update\\_webjet7.jsp\)](http://localhost/admin/update/update_webjet7.jsp) for database conversion. If you don't get the message, the installation database is already ready in the new format, continue [by setting the rights](#).



In the conversion page, start at the end [password hashing \(http://localhost/admin/update/update\\_passwords.jsp\)](http://localhost/admin/update/update_passwords.jsp) to switch password storage mode to secure `hash`.

We also recommend deleting `STAT` tables as shown in the page (not necessary), for example by inserting commands into </admin/updatedb.jsp> (<http://localhost/admin/updatedb.jsp>).

```

DROP TABLE stat_browser;
DROP TABLE stat_country;
DROP TABLE stat_site_days;
DROP TABLE stat_site_hours;
DROP TABLE stat_doc;
DROP TABLE stat_views;

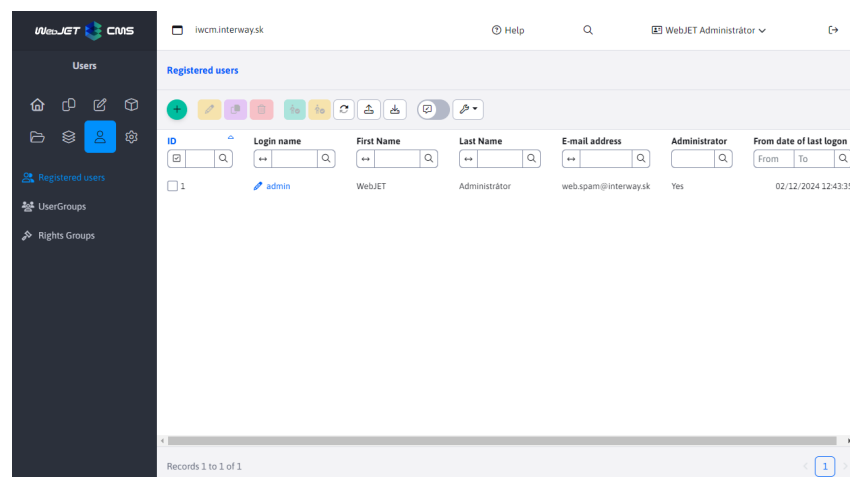
```

sql

Close the tab in which you have the conversion.

## 2.6. Setting rights

In the original window, go to [Users -> List of users \(http://localhost/admin/v9/users/user-list/\)](http://localhost/admin/v9/users/user-list/), using the left menu navigation.



Open editing **Admin** Users. When the window appears, check the tabs **Personal data** a **Contact details** your details.

In the charts **Rights** enable the necessary rights. At a minimum, add rights:

- Configuration
- Configuration - view all variables

After setting the rights, log out to apply the new rights and log in again. Once logged in, go to [Settings/Configuration](http://localhost/admin/v9/settings/configuration/) (<http://localhost/admin/v9/settings/configuration/>) and set the following conf. variables:

- If the server is in an InterWay environment, or is located behind a proxy/load balancer, set the variable `serverBeyondProxy` to the value of `true`. In this mode, WebJET expects the IP address of the visitor in the HTTP header `x-forwarded-for` and the protocol used in `x-forwarded-proto`.
- You can set the variable `logLevel` to the value of `debug` for more detailed logging.
- We recommend setting the variable `webEnableIPs` to a list of IP address prefixes from which you will access the site before launching (e.g. 127.0.0.1,10.,192.168.,195.168.35.4,195.168.35.5).

Next, follow the instructions for [template settings](https://docs.webjetcms.sk/latest/en/frontend/setup/README) (<https://docs.webjetcms.sk/latest/en/frontend/setup/README>).

### 3. Setting the license number

The license number for WebJET is entered in the Settings/Configuration section in the conf. variable `license`. It is possible to have additional licenses for additional domains (e.g. when using the Enterprise license) in the conf. variable `licenseDomains` where each additional domain license is entered on a new line.

### 3.1. Incorrect licence number

If WebJET contains a license number with an expired expiration date, an incorrect domain name, or the license number is entered incorrectly, the option to update the license number will be displayed in the server logs after logging in to the administration at startup:

Log

```
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 - -----
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 - WebJET initializing, root:
/www/tomcat/webapps/webjet
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 -
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 - VERSION: 9
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 - Checking database connection:
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 - Database connection: [OK]
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 - INIT (db): license=xxxxxxxxxxxxxxxxxxxxxxxxxxxx
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 - License is valid until: 01.06.2022 23:59:59
[webjet] [s.i.i.InitServlet] [INFO] [0] 2023-10-02 09:27:30 - ERROR: License is out of date, please contact
InterWay (www.interway.sk)
for new license.
```

If the option to enter a license number does not appear automatically when you open the administration section, open the address `/wjerrorpages/setup/license`.

**Server license is not valid.**

Please contact [InterWay, a.s. Bratislava](mailto:sales@interway.sk) at:

InterWay, a.s. Bratislava  
 Stará Vajnorská 21  
 831 04 Bratislava  
 Slovakia  
 tel: +421 2 3276 6880, +421 905 222 226  
 email: [sales@interway.sk](mailto:sales@interway.sk)

Login Name:

Password:

License key:

English

Enter your login credentials to verify your administrative privileges and your new license key. Click OK to save. You will see a confirmation of the save:

**License successfully changed.**

The license was successfully changed. Wait approximately 5 minutes and then [log in to the admin area](#).

Thank you for using WebJET. If you have any questions or comments, please [contact us](#).

If the application server does not restart automatically, restart the application server. The restart will use the specified license number.

## 3.2. Enter the license number directly into the database

If you cannot enter the license number through the web interface, you can use the following SQL statement directly on the database server:

```
UPDATE _conf_ SET value='xxx' WHERE name='license';
```

sql

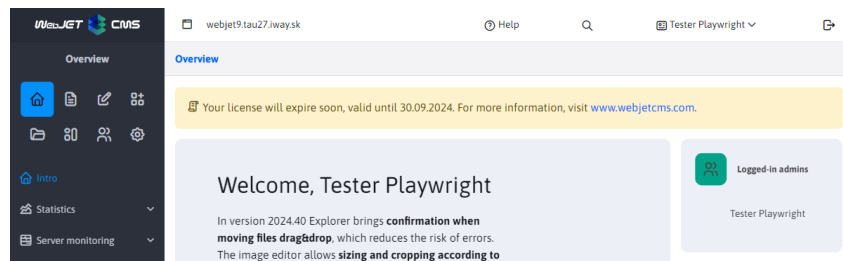
for Oracle use the notation:

```
UPDATE webjet_conf SET value='xxx' WHERE name='license';
```

sql

## 3.3. Expiry of the licence

WebJET can automatically detect the upcoming expiration of your license. If it detects that your license will expire in 2 (or less) months, it will display a warning in the home page.



## 4. Basic configuration

The most commonly used [configuration variables](https://docs.webjetcms.sk/latest/en/admin/setup/configuration/README) (<https://docs.webjetcms.sk/latest/en/admin/setup/configuration/README>).

### 4.1. Logging in

- `logLevel` - basic logging level, can have a value of `debug` for detailed logging, or `normal` for production deployment.
- `logLevels` - list of java packages with logging level (each on a new line), e.g:

```
sk.iway=DEBUG
sk.iway.iwcm=WARN
org.springframework=WARN
```

### 4.2. Sending emails

To send emails you need to set up the SMTP server correctly:

- `smtpServer` - SMTP server address for sending emails.
- `smtpUseSSL` - by setting it to `true` activate the use of SSL.
- `smtpUseTLS` - TLS authentication - if port 587, `smtpUseTLS` must be `true`.
- `smtpTLSVersion` - version `TLS` for smtp connection.
- `smtpUser` - login name.
- `smtpPassword` - password.
- `smtpPort` - port for connection to the SMTP server.
- `useSMTPServer` - disable email forwarding (e.g. for cluster nodes that do not have an SMTP server available).
- `smtpConnectionTimeoutMillis` - the number of milliseconds to wait for the SMTP connection to be established.

- `emailProtectionSenderEmail` - set the email address to be used as the sender's email for all emails if SMTP is not set `OPEN RELAY` . The typical value is `noreply@domena.sk` . When set up, the email address entered is set in the header of each email `FROM` and the originally set value of `z FROM` is set to `REPLY-TO` .

For bulk email you can still set up:

- `dmailWaitTimeout` - speed of sending emails from bulk email in milliseconds. By default it is set to 5000, which means that an email is sent once every 5 seconds. If you lower the value then the web server and SMTP server will be more loaded when sending emails. The value will only take effect after a server restart.
- `dmailBadEmailSmtReplyStatuses` - A comma-separated list of expressions returned from the SMTP server for which the email will not attempt to be sent again.

### 4.2.1. Setting up Amazon SES

For bulk email, we recommend using [Amazon Simple Email Services/SES \(https://aws.amazon.com/ses/\)](https://aws.amazon.com/ses/) for better email delivery. Originally WebJET CMS used API access, which was activated by setting the conf. variable `useAmazonSES` to the value of `true` , but currently it is already in use [standard SMTP protocol \(https://docs.aws.amazon.com/ses/latest/dg/send-email-smtp.html\)](https://docs.aws.amazon.com/ses/latest/dg/send-email-smtp.html) in Amazon SES:

- Select [the address of the SMTP server \(https://docs.aws.amazon.com/general/latest/gr/ses.html\)](https://docs.aws.amazon.com/general/latest/gr/ses.html) for your region and set it in the conf. variable `smtpServer` , e.g. `email-smtp.eu-west-1.amazonaws.com` . The tables are scrollable on the page, only the US region is visible at first, don't be afraid to scroll the table.
- Create [login details \(https://docs.aws.amazon.com/ses/latest/dg/smtp-credentials.html\)](https://docs.aws.amazon.com/ses/latest/dg/smtp-credentials.html) to the SMTP server and set them to conf. variables `smtpUser` a `smtpPassword` , select Encrypt for the password.
- On the page [Amazon SES \(https://console.aws.amazon.com/ses/\)](https://console.aws.amazon.com/ses/) in the SMTP settings section for the selected region you can also see the individual ports through which it communicates, typically it is necessary to enable communication to port 587 on the firewall.
- For a new project, after testing, ask for an increase in email sending limits, they are set low by default.
- Set conf. variable `smtpUseTLS` at `true` .
- V [Amazon SES \(https://console.aws.amazon.com/ses/\)](https://console.aws.amazon.com/ses/) in the Identities section you need to verify the domain identity and set `DKIM` Keys.
- Delete conf. variable `useAmazonSES` if you have it set (for older projects where API access was originally used).
- Restart the application server.
- Try sending an email, verify in the email headers that it was actually sent via Amazon SES.

Sending via Amazon SES sets `DKIM` headers and ensure higher deliverability of emails.

## 4.3. Cluster

In the case of a cluster installation, WebJET needs to know that it is running in a cluster and has to synchronize the internal memory cache.

Can run in fashion `auto` when it doesn't need a list of nodes, or in a mode where it has them listed exactly.

### 4.3.1. Auto mode

The easiest is to run in auto mode, conf. variable `clusterNames` set to `auto` and restart the server. In this case, the node name/ `nodu` the first found value of `z` is used:

- ENV variable `HOSTNAME` server
- call value `InetAddress.getLocalHost().getHostName()` - the domain name of the computer
- call value `InetAddress.getLocalHost().getHostAddress()` - IP address of the computer
- Value `"auto-"+Tools.getNow()`

The value is truncated to the first 16 characters. If the variable is `clusterHostnameTrimFromEnd` set to `true` , the trailing 16 characters are used (e.g. kubernetes creates `hostname` with a random value at the end).



### 4.3.2. Exact list of nodes

If you have a stable configuration of running nodes/ `nodes` set the conf. variable:

- `clusterNames=node1,node2,node3` - comma separated list of nodes from 1 to N

You need to externally define individual nodes `ID` node, you can't do it via Settings->Configuration, because all nodes would have the same name.

It is recommended to set the value via the parameter `-DwebjetNodeId=1`, or otherwise via [external configuration](#).

### 4.3.3. Other conf. variables

- `clusterRefreshTimeout` - node synchronization interval in ms, default `5000ms`. For production, a value of 60000 (1 minute) is usually sufficient to reduce the load.
- `clusterMyNodeType` - node type - `full` = administration and presentation part, `public` = only the presentation part without administration.
- `senderRunOnNode` - if set to a non-empty value contains a list of cluster nodes on which bulk email sending is triggered (e.g. node1 or node1,node2). It is recommended to trigger the send only on the administration node.

## 4.4. Primary key generator

For some parts the primary key generator has historically been used, the following values can be set:

- `pkeyGenIncrement` - the value by which it increases.
- `pkeyGenOffset` - shift value for the cluster.
- `pkeyGenBlockSize` - the size of the block selection for the primary key generator. By default set to 10, for server with high load we recommend to set it to a higher value (100 - 1000).

To avoid conflicts in the cluster configuration, the value `pkeyGenOffset` for nodeshift. E.g. value `pkeyGenIncrement` is set to 5 and `offset` to 0-5 for individual nodes. In mode `auto` clustra is the automatically set value `pkeyGenBlockSize=1` to always read the last value from the database. This has a slight impact on server performance.

## 4.5. Licences

Some libraries may need to purchase additional licenses for their use:

- `amchartLicense` - license number for the library [amCharts](https://www.amcharts.com) (<https://www.amcharts.com>) to display charts, after setting the license key, the amCharts logo will not be displayed in the chart.

## 5. External configuration

---

WebJET is configured by default directly in the administration in the Settings/Configuration section. However, sometimes it is necessary to configure variables differently for individual cluster nodes, or to transfer configuration from external variables (e.g. different for PROD and TEST servers).

WebJET also supports setting configuration variables from `Java System Properties` also from environment variables `Enviroment Variables`.

## 5.1. System variables

System variables are set for each running Java process (application server). They can be **differently for each application server running within a single operating system**.

To take the value of the system variable `System.getProperty` it is necessary to prefix the variable name with the expression `webjet.`. You get the variables into WebJET by setting the `-D` parameter of the java process.

On Linux OSes you usually edit the file `/etc/conf.d/tomcat_XXX` or `/etc/default/tomcat_XXX` where you add the desired variable as:

```
#WebJET: vypnutie dmail sendera (aby sa neposielali mailu duplicitne)
JAVA_OPTS="$JAVA_OPTS -Dwebjet.disableDMailSender=true"
#WebJET: ID nodu clustra
JAVA_OPTS="$JAVA_OPTS -DwebjetNodeId=2"
```

sh

The above entry sets the configuration variable `disableDMailSender` to the value of `false`.

**COMMENT:** Note the setting of the cluster node ID directly by setting the variable `webjetNodeId` that does not use a prefix `webjet.`. `webjetNodeId` is not a configuration variable, but an instruction to set configuration variables:

- `clusterMyNodeName` - to the value of `nodeX`
- `pkeyGenOffset` - to the value of `X`

Where `X` is the value set via `webjetNodeId`.

In Windows, system variables are set in the program `Configure Tomcat` in the tab `Java`. You can set the above variables by adding them to the end of the text area `Java Options`:

```
-Dwebjet.disableDMailSender=true
-DwebjetNodeId=2
```

## 5.2. Enviroment variables

Environment variables are usually set at the operating system level and **are common to all application servers running on a given operating system**.

They are set with a prefix `webjet_` since the name of an environment variable cannot contain the period character. In java code, they are taken as `System.getenv`. On Linux OS in `shell` set variables like:

```
#pre csh pouzite namiesto export setnev
export webjet_disableDMailSender=true
export webjetNodeId=2
```

sh

**We recommend using environment variables when containerizing**, because they can be set by the standard way by a triggered container.

## 5.3. Context parameter

A suitable place to set the variable is also `<Parameter>` v `<Context>` element in `server.xml` for Tomcat. The advantage is that you can set the variable for each host separately. They are set with the prefix `webjet_`

```
<Host name="meno.domena.sk" debug="0" appBase="webapps2" unpackWARs="false" autoDeploy="false">
  <Alias>alias.domena.sk</Alias>
  <Context path="" docBase="../webapps/webjet" reloadable="true" debug="0" swallowOutput="true">
    <Resources allowLinking="true" />
    <Parameter name="webjet_meno-premennej" value="hodnota" override="true" />
  </Context>
  <Valve className="org.apache.catalina.valves.RemoteIpValve"
    remoteIpHeader="x-forwarded-for" protocolHeader="x-forwarded-proto" />
</Host>
```

xml

## 5.4. Empty value

When entering a value via `-Dwebjet.` or `ENV` value is used only if it is not empty. To specify an empty value, enter the character `-`, this will be replaced with an empty value.

## 5.5. External database connection

The database connection is set in WebJET in the file `WEB-INF/classes/poolman.xml`. Sometimes it is advisable to change the database connection parameters according to the environment (PROD, TEST), or security does not allow to have the database password written in the file.

WebJET supports setting database connection parameters from system/environment variables. File `poolman.xml` must exist, but the values can be empty:

```
<?xml version="1.0" encoding="UTF-8"?>
<poolman>
  <datasource>
    <dbname>iwcm</dbname>
    <jndiName>jndi-iwcm</jndiName>
    <driver>oracle.jdbc.driver.OracleDriver</driver>
    <url></url>
    <username></username>
    <password></password>
    <minimumSize></minimumSize>
    <maximumSize></maximumSize>
  </datasource>
</poolman>
```

xml

You can then set individual values via system or environment variables. The following variables are supported:

- `webjetDbDriver`
- `webjetDbUserName`
- `webjetDbPassword`
- `webjetDbUrl`
- `webjetDbMinimumSize`
- `webjetDbMaximumSize`

only the values that are set are downloaded. So you can combine the settings `minumumSize` v `poolman.xml` and then the variable `webjetDbMinimumSize` you don't have to set it via an environment variable.

**TIP:** for setting the variable v `JAVA_OPTS` don't forget to add a parameter to the variable menu `-D` .

If you need to define multiple database connections for a connection other than `iwcm` it is possible to use a suffix in the variable name with the value `_dbname` that is, for example `webjetDbUserName_ip_data_jpa` .

### 5.5.1. Using another file

WebJET also allows you to use other `poolman.xml` file as default. Just use `JAVA_OPTS` set parameter `-DwebjetPoolmanPath=/poolman-local.xml` with the path to another poolman.xml file. The specified name is first searched for directly as a file on disk (specified as an absolute path), if not found as a file in the directory `WEB-INF/classes` .

Settings `-DwebjetPoolmanPath=/poolman-local.xml` is global for all of Tomcat, but WebJET will only use the file if it exists, otherwise it will use the standard `poolman.xml` .

If you need to set a different file for just one `Host` Tomcat, you can use the Context parameter `webjetDbname` in the tomcat/server.xml file. If it contains a path ending in .xml it is not used as the database connection name, but as the filename:

```
<Context reloadable="false" path="" docBase="/www/tomcat/webapps/webjet/" allowLinking="true"
swallowOutput="true">
  <Parameter name="webjetDbname" value="/poolman-acc.xml" override="true"/>
</Context>
```

xml

### 5.5.2. Switching configuration between environments

If you need to switch the database connection according to the environment you can in `poolman.xml` define multiple database connections. Primary `iwcm` connection can be set to dev database, at the same time you add a new connection named `acc` for connection to `ACC` environment.

Subsequently on `ACC` environment in the Tomcat configuration in `server.xml` you can configure WebJET so that instead of `iwcm` connections used `acc` connection by setting the Context parameter `webjetDbname` :

```
<Context reloadable="false" path="" docBase="/www/tomcat/webapps/webjet/" allowLinking="true"
swallowOutput="true">
  <Parameter name="webjetDbname" value="acc" override="true"/>
</Context>
```

xml

## 6. Spam protection

WebJET includes integrated SPAM protection, which limits the rate of requests and their number to avoid overloading the server. Such protection is also referred to as `Rate Limiting` , i.e. limiting the number of requests.

The protection is tied to the visitor's IP address, so the limits apply to each IP address separately. It is therefore necessary to have the visitor's IP address retrieval set up correctly. For example, if the application server is predefined before the `Load Balancer/Proxy` it is necessary to set the configuration variable `serverBeyondProxy` to the value of `true` and on `Load Balancer/Proxy` set HTTP headers `x-forwarded-for` with the IP address of the visitor and `x-forwarded-proto` with the value of the protocol used `http/https` . You can verify the correctness of the IP address in [audit \(https://docs.webjetcms.sk/latest/en/sysadmin/audit/README\)](https://docs.webjetcms.sk/latest/en/sysadmin/audit/README) in the IP address column when you perform an action, such as logging in or submitting a form.

**Warning:** if you are using a cluster, the limiting happens separately on each application server/node of the cluster - data is not shared between nodes.

## 6.1. Basic settings

The following configuration variables are used to set the basic limiting of the number of requests:

- `spamProtectionHourlyLimit` - The maximum number of requests from one IP address per hour, set to 20 by default.
- `spamProtectionTimeout` - The minimum number of seconds between two requests from the same IP address, set to 30 seconds by default.
- `spamProtectionIgnoreFirstRequests` - The number of first requests from an IP address that are not capped, set to 0 by default, i.e. no capping exception.

### 6.1.1. Settings for module/application

Limits can be modified specifically for certain modules/applications by adding `-appName` to the end of the configuration variable, e.g. `spamProtectionHourlyLimit-appName` or `spamProtectionTimeout-appName`. The values can be set for the following applications:

- `dmail` - [bulk email](https://docs.webjetcms.sk/latest/en/redactor/apps/dmail/form/README) (https://docs.webjetcms.sk/latest/en/redactor/apps/dmail/form/README) - limits the number of registrations/unsubscribes from the bulk email.
- `form` - [forms](https://docs.webjetcms.sk/latest/en/redactor/apps/form/README) (https://docs.webjetcms.sk/latest/en/redactor/apps/form/README) and sending feedback from the administration homepage.
- `forum` - [discussion forum](https://docs.webjetcms.sk/latest/en/redactor/apps/forum/README) (https://docs.webjetcms.sk/latest/en/redactor/apps/forum/README), limits the number of posts added.
- `HtmlToPdfAction` - generation of PDF documents.
- `inquiry` - [survey](https://docs.webjetcms.sk/latest/en/redactor/apps/inquiry/README) (https://docs.webjetcms.sk/latest/en/redactor/apps/inquiry/README), limits the number of votes in the poll.
- `passwordSend` - [sending a forgotten password](https://docs.webjetcms.sk/latest/en/redactor/admin/password-recovery/README) (https://docs.webjetcms.sk/latest/en/redactor/admin/password-recovery/README), limits the number of times a forgotten password can be sent.
- `qa` - [questions and answers](https://docs.webjetcms.sk/latest/en/redactor/apps/qa/README) (https://docs.webjetcms.sk/latest/en/redactor/apps/qa/README), limits the addition of new questions.
- `quiz` - [questionnaires](https://docs.webjetcms.sk/latest/en/redactor/apps/quiz/README) (https://docs.webjetcms.sk/latest/en/redactor/apps/quiz/README), limits the number of questionnaires sent.
- `search` - [Search](https://docs.webjetcms.sk/latest/en/redactor/apps/search/README) (https://docs.webjetcms.sk/latest/en/redactor/apps/search/README), limits the number of searches.
- `ThumbServlet` - [preview images](https://docs.webjetcms.sk/latest/en/frontend/thumb-servlet/README) (https://docs.webjetcms.sk/latest/en/frontend/thumb-servlet/README), limits the number of times a new preview image is generated, the image is saved to disk and used for further requests.
- `userform` - [new user registration](https://docs.webjetcms.sk/latest/en/redactor/zaheslovana-zona/README) (https://docs.webjetcms.sk/latest/en/redactor/zaheslovana-zona/README), limits the number of new registrations and modifications to a user's profile in the password-protected zone.

By default, the following values are already adjusted:

- `spamProtectionHourlyLimit-ThumbServlet` - number of requests for generation [image thumbnails](https://docs.webjetcms.sk/latest/en/frontend/thumb-servlet/README) (https://docs.webjetcms.sk/latest/en/frontend/thumb-servlet/README), `/thumb` set to 300.
- `spamProtectionTimeout-ThumbServlet` - set to `-2`, which means that limiting the spacing between the requirements for `/thumb` is not applied, this is because there can be several such images in the page at the same time and HTTP requests are executed in parallel.
- `spamProtectionHourlyLimit-search` - The number of search requests through the search application, set to 200.
- `spamProtectionTimeout-search` - value reduced to 10 to allow faster search/advance to the next page of results.

## 6.2. Exceptions

The protection can be partially disabled by setting the following variables:

- `spamProtectionDisabledIPs` - a comma-separated list of IP address beginnings (or a character `*` for all) for which spam protection is disabled.

Reset status is possible in the application [Deleting data - Cache objects \(https://docs.webjetcms.sk/latest/en/sysadmin/data-deleting/README\)](https://docs.webjetcms.sk/latest/en/sysadmin/data-deleting/README), where the list of the number of calls and intervals for all IP addresses is reset using the action Delete all cache objects.

## 7. Migration between environments

---

Basic information that needs to be done when migrating between environments, e.g. production and development environments.

### 7.1. Configuration variables

Check the following conf. variables:

- `cloudStaticFilesDir` - the path to the static files directory, it needs to be set correctly on the environment (or deleted on the environment if external static files are not used).
- `enableStaticFilesExternalDir` - enabling/disabling the use of an external directory for static files.
- `smtpServer` - SMTP server address + variables `smtpUsername`, `smtpPassword`, `smtpPort`, `smtpUseTLS` and the like.
- `emailProtectionSenderEmail` - set the email address to be used as the sender's email, it can be different for each environment, it must be enabled on the SMTP server.
- `proxyHost/proxyPort` - proxy settings
- `webEnabledIPs` - the list of IP addresses from which the web is accessible, on the DEV environment set by default to `#localhost,127.0.0.1,10.,192.168.,#interway,85.248.107.8,195.168.35.4,62.168.118.90,62.168.118.67,#klient,`
- `adminEnabledIPs` - the list of IP addresses from which it is possible to access the administration, by default set to `#localhost,127.0.0.1,10.,192.168.,#interway,85.248.107.8,195.168.35.4,#klient,`
- `multidomainAdminHost` - if set to verify domain, it is used to set the domain for the CMS on a multi-domain installation.
- `serverBeyondProxy` - Set the location of the application server behind the proxy/load balancer.
- `logLevel` - on production to leave on `normal`, on the development environment you can change to `debug`.

Check all other conf. variables and consider their impact on the project and the appropriateness of the setting.

### 7.2. Domain addresses

If the site uses `multidomain` you need to set up the domains correctly. When migrating between environments, check all root directories and set the correct domain names. Also check the option Change domain redirects, but if the domain is set in multiple root directories (e.g. both en and en) check this only when changing the domain for the first time.

## 7.3. Users (when migrating to a DEV environment)

To avoid accidentally sending a request for approval of a website from the developer environment, I recommend changing the email addresses of existing accounts to appropriate developer accounts.

Verify user settings `admin`, set its permissions on all directories (e.g. by setting the approval mode to none), permissions on modules, and set an appropriate password.

## 7.4. Background tasks

Check the background tasks, especially the set task URLs. On the DEV environment, delete unnecessary tasks if necessary, on the PROD environment in case of a cluster, check the settings of the node on which the task is to be executed.

## 7.5. Data deletion (when migrating to DEV environment)

To prevent the local database on the DEV environment from becoming unnecessarily large, we recommend that you perform a data deletion:

- keep statistics only for the last 3 months
- delete emails completely (ideally via `DELETE * FROM emails`)
- delete server monitoring

## 7.6. Other

Consider the specific features of the project and its settings and apply them. Consider whether it contains any more specific settings, e.g. in the directory structure (free fields) and so on.

## 7.7. Logos

Check the application server logs and fix any errors during startup (e.g. due to bad autoupdate.xml, etc.).

# 8. MultiWeb

---

## 8.1. Home

MultiWeb is a special type of multi-user/multi-tenant installation in a single WebJET. The domains are externally separate and each pretends to be a separate WebJET CMS installation.

The installation includes a control domain (the first one established), through which elevated permissions are available (e.g. for configuration, translation keys, etc.). Applications that impact all domains are not available in individual tenant domains.

## 8.2. Restrictions

MultiWeb installations contain the following restrictions:

- Web pages
  - Media groups - only media groups with folder rights set to folders in the current domain are displayed and editable; newly added media groups are automatically given rights to the root folders in the domain. You can manage global media groups in the control domain without specifying the View For restriction.
- Applications
  - Domain redirects - only domain redirects pointing to the current domain can be managed.
- Templates

- The list displays templates that have access set to the folders of the currently displayed domain, when a new template is created that would have no access set, access is automatically set to the root folders of the current domain.
- Template Groups - displays the template groups used in the templates.
- Users
  - Each domain has its own list of users.
  - For a user who has no restrictions set on the tree structure of Web site folders, a list of domain root folders is set internally when checking permissions.

## 8.3. Applications available only in the management domain

The following applications are only available in the management domain:

- Home
  - Audit
  - Server monitoring
- Applications
  - GDPR - Regular expressions
  - GDPR - Data erasure
  - Bulk Email - Domain Limits
- Users
  - User groups
  - Groups of rights
- Control panel
  - Configuration
  - Translation keys
  - HTTP headers
  - Data deletion
  - Automated tasks
  - WebJET CMS update
  - Restart

If you need to add some rights to a user in the domain, you can add a conf. variable `multiwebSpecialPerms-USERID` A comma-separated list of additional rights that the user is to receive.

## 9. Configuration

---

The following requirements must be met for proper operation:

- [Setting up templates \(https://docs.webjetcms.sk/latest/en/frontend/setup/README\)](https://docs.webjetcms.sk/latest/en/frontend/setup/README) for managing multiple domains
- The first domain is called control domain, it should not contain a real web page, it is used for setting global parameters.
- We recommend creating a domain alias for each domain. You create it in the configuration as a key `multiDomainAlias:DOMAIN-NAME` with a suitable domain name value without spaces, www and suffixes, e.g. `interway`. Domain alias allows you to prepare a website on a working domain and then move it to a production domain. The specified domain alias will be used to search for template files and customized application files.



## 9.1. Setting up a new domain

Currently, you need to create a new domain using the SQL statement below. The expression `DOMAIN-NAME` replace with the domain name. The login name is `Heslo1`. Once added, you need to either delete the cache via the control domain in Control Panel->Cache Objects by clicking on delete all, or restart the application server.

sql

```
INSERT INTO `groups` (`group_name`, `internal`, `parent_group_id`, `navbar`, `default_doc_id`, `temp_id`,
`sort_priority`, `password_protected`, `menu_type`, `url_dir_name`, `sync_id`, `sync_status`, `html_head`,
`logon_page_doc_id`, `domain_name`, `new_page_docid_template`, `install_name`, `field_a`, `field_b`, `field_c`,
`field_d`, `logged_menu_type`, `link_group_id`)
VALUES
('DOMAIN-NAME', 0, 0, 'DOMAIN-NAME', -1, 1, 0, '', 2, '/', 1, 1, '', -1, 'DOMAIN-NAME', -1, '', '', '', '',
'', -1, -1);

INSERT INTO `groups` (`group_name`, `internal`, `parent_group_id`, `navbar`, `default_doc_id`, `temp_id`,
`sort_priority`, `password_protected`, `menu_type`, `url_dir_name`, `sync_id`, `sync_status`, `html_head`,
`logon_page_doc_id`, `domain_name`, `new_page_docid_template`, `install_name`, `field_a`, `field_b`, `field_c`,
`field_d`, `logged_menu_type`, `link_group_id`)
VALUES
('System', 1, 0, 'System', -1, 1, 10, NULL, 0, 'system', 0, 1, '', -1, 'DOMAIN-NAME', -1, NULL, '', '', '',
'', -1, -1);

INSERT INTO `users` (`title`, `first_name`, `last_name`, `login`, `password`, `is_admin`, `user_groups`,
`company`, `adress`, `city`, `email`, `PSC`, `country`, `phone`, `authorized`, `editable_groups`,
`editable_pages`, `writable_folders`, `last_logon`, `module_perms`, `disabled_items`, `reg_date`, `field_a`,
`field_b`, `field_c`, `field_d`, `field_e`, `date_of_birth`, `sex_male`, `photo`, `signature`, `forum_rank`,
`rating_rank`, `allow_login_start`, `allow_login_end`, `authorize_hash`, `fax`, `delivery_first_name`,
`delivery_last_name`, `delivery_company`, `delivery_adress`, `delivery_city`, `delivery_psc`,
`delivery_country`, `delivery_phone`, `position`, `parent_id`, `password_salt`, `domain_id`)
VALUES
('', 'DOMAIN-NAME', 'Admin', 'admin', 'ead38302881122ab70592e113663c475', 1, NULL, '', '', '',
'info@webjet.eu', '', '', '', 1, '', '', '', '2014-08-07 14:43:29', NULL, NULL, '2014-02-05 14:55:46', '', '',
'', '', '', NULL, 1, '', '', 0, 0, NULL, NULL, NULL, '', '', '', '', '', '', '', '', '', 0, NULL, 230);
```