Kravmotoren

Developer Guide

**Date**: 18.09.2017

**Author**: BSG

Content

[1 Introduction 3](#_Toc493498080)

[1.1 Prerequisites 3](#_Toc493498081)

[1.2 The MySQL prerequisite 3](#_Toc493498082)

[2 Compiling and running Kravmotoren 3](#_Toc493498083)

[2.1 Login 4](#_Toc493498084)

[3 Integrations 4](#_Toc493498085)

[4 Internationalization (i18n) 4](#_Toc493498086)

[5 Frameworks 4](#_Toc493498087)

# Introduction

This document is intended for developers, who are going to work on Kravmotoren. For details on what Kravmotoren is, and how it functions, consult the other documents that came with Kravmotoren.

## Prerequisites

Kravmotoren is a Java application, and it uses the following tools, which must be available to build and work with Kravmotoren.

* Java JDK 8+ (OpenJDK or Oracle JDK will work fine)
* Apache Maven 3+
* MySQL 5.5+ installed

As a developer, you will most likely be using an IDE like Eclipse or IntelliJ. Kravmotoren uses Project Lombok for generating a range of boilerplate code, and Project Lombok has made a plugin available for most common Java IDEs, so the generated code is made available for auto-completion in the IDEs.

<https://projectlombok.org/>

Without installing the Project Lombok plugin in your IDE, you will not be able to work reliable with the Kravmotoren source in your IDE.

## The MySQL prerequisite

Kravmotoren uses MySQL as database storage, and both during development, and when running tests, a MySQL database server must be available. Using an in-memory database like H2 might have made for faster/easier setup of a development environment, but at the cost of a difference between development and production.

Installing a MySQL server is probably the easiest approach, but if this is not possible, then running MySQL as a local Docker container is a suitable substitute

<https://hub.docker.com/_/mysql/>

# Compiling and running Kravmotoren

During development, the project can be started from within your IDE, simply by running the main method found in the Application.java file.

From the command-line, the project can be compiled and executed using Maven.

Note that the project has 2 sub-modules (ui and webjar), and both need to be compiled, so from the root folder, execute the following command

$ mvn clean install

This will compile the source in both modules, and run all tests in the project. The tests in the ui module contains some selenium tests (UI testing). These tests rely on a chromedriver binary that is bundled with the source. The binary is for Linux. If you are developing on another platform, you must download a chromedriver binary for your chosen platform, and replace the binary in the source folder

<https://sites.google.com/a/chromium.org/chromedriver/>

Also, as stated in chapter 1.2, you will need a running MySQL database, with the needed databases created for the tests to run (username, password and database can be configured in (ui/src/test/resources/test.properties) for tests and (ui/config/application.properties) for development.

Once the source has been compiled, you can start the Rollekatalog from the ui module

$ cd ui

$ mvn spring-boot:run

The application will bind to port 8099 and require HTTPS to access it, so it should be available on the following address

<https://localhost:8099/>

## Login

Kravmotoren uses SAML to deal with user login, and the development configuration is pre-configured to target Digital Identitys development AD FS for login. You can replace this configuration with your own SAML Identity Provider (see ui/config/application.properties for details).

Working knowledge about SAML and Identity Providers is assumed.

# Integrations

Besides depending on an external SAML Identity Provider for login, Kravmotoren also depends on an external SMTP relay, that can forward notifications emails to relevant recipients.

For production Kravmotoren uses Amazon SNS (simple notification service) as the SMTP relay, but for development purposes, it might be prudent to use a “stubbed” MailSenderService implementation (one can be found with the test classes if needed).

# Internationalization (i18n)

Kravmotoren supports internationalization, and all texts are read from a message resource bundle. Two translations currently exist, Danish and English, though it should be noted that only the Danish translations have gone through quality control, and the English translations are intended for internal development use only (for now).

# Frameworks

Kravmotoren makes heavy use of the Spring Framework, especially the following components of the Spring Framework suite

* Spring Boot. The convention-over-configuration approach to using Spring.
* Spring Data JPA. The abstraction layer for accessing databases.
* Spring MVC. The implementation of the MVC UI pattern.
* Spring Security. The implementation of a full authentication and authorization stack.
* Spring Actuator. Production-grade monitoring of the application.

For the frontend, a combination of Bootstrap, JQuery and Thymeleaf makes up most important frameworks.