

Architecture

[frontend](#), [architecture](#)

The application is divided into 2 technological lines: backend (java) and frontend (javascript). There is a REST interface between the two.

Minimum executable application

The minimal executable application always contains the app + core modules.

Core module

Contains services, managers, components and individual core pages. More in [modularity](#) and in the next chapter with layers.

App module

= Application, executable application. Connects modules with environment configuration.

Modularity

[modularity](#), [frontend](#), [architecture](#), [installation](#)



TODO: Introduction

Installation/ module update

Installation/ module update is done by copying the module into the application libraries **<idm.war>/WEB-INF/lib**. Modules can also be included directly in the project dependencies (pom.xml modul app).

You can see a list of all installed modules and their status (enabled/disabled) in the Modules agenda in the Settings.

Modules can be enabled/disabled by setting **idm.pub.<identifier_of_module>.enabled** as true, or false.

Module-specific database scripts

Each module can contain its own set of [database scripts](#), which are applied to the application database scheme at the start of the application (in the phase after initializing the data connection, or rather after initializing the hibernate context) ⇒ **the modules are automatically initializing and updating the database scheme themselves.**

Example module

The [example module](#) was created for module demonstration and, at the same time, as a template (skeleton) for creating a module with pre-filled configurations.

From:

<https://wiki.czechidm.com/> - **CzechIdM Identity Manager**

Permanent link:

<https://wiki.czechidm.com/devel/documentation/adm/architecture>

Last update: **2019/03/04 14:00**

