

# Application Profiles

## Contents

1. [Profiles](#)
2. [Routing](#)
3. [web servers](#)
4. [databases](#)
5. [ClojureScript](#)
6. [service API](#)
7. [miscellaneous](#)

## Profiles

Running `lein new luminus myapp` will create an application using the default profile template. However, if you would like to attach further functionality to your template you can append profile hints for the extended functionality.

## Routing

+reitit - adds [Reitit](#) Clojure/Script router support

## web servers

Luminus defaults to using the [Immutant](#) webserver, the following alternative servers are supported:

- +aleph - adds [Aleph](#) server support to the project
- +jetty - adds [Jetty](#) support to the project
- +http-kit - adds the [HTTP Kit](#) web server to the project

Build Tool:

## Topics

- Your First Application
- REPL Driven Developme
- **Application Profiles**
  - HTML Templating
  - Static Assets
  - ClojureScript
  - Routing
  - RESTful Services
  - Request types
  - Response types
  - Websockets
  - Middleware
  - Sessions and Cookies
  - Input Validation
  - Security
  - Component Lifecycle
  - Database Access
  - Database Migrations
  - Logging
  - Internationalization
  - Testing
  - Server Tuning
  - Environment Variables
  - Deployment
  - Useful Libraries
  - Sample Applications
  - Upgrading
  - Clojure Resources

## Books

## databases

- +h2 - adds `db.core` namespace and H2 db dependencies
- +sqlite - adds `db.core` namespace and SQLite db dependencies
- +postgres - adds `db.core` namespace and add PostgreSQL dependencies
- +mysql - adds `db.core` namespace and add MySQL dependencies
- +mongodb - adds `db.core` namespace and MongoDB dependencies
- +datomic - adds `db.core` namespace and Datomic dependencies

## ClojureScript

- +cljs - adds [ClojureScript](#) support
- +hoplon adds ClojureScript support with [Hoplon](#) to the project
- +reagent - adds ClojureScript support with [Reagent](#)
- +re-frame - adds ClojureScript support with [re-frame](#)
- +kee-frame - adds ClojureScript support with [kee-frame](#)

## service API

- +graphql - adds GraphQL support using [Lacinia](#)
- +swagger - adds support for [Swagger-UI](#) using the [compojure-api](#) library
- +service - create a service application without the front-end boilerplate such as HTML templates

## miscellaneous

- +boot - uses [Boot](#) as build tool and creates `build.boot` instead of `project.clj`
- +auth - adds [Buddy](#) dependency and authentication middleware
- +auth-jwe - adds [Buddy](#) dependency with the [JWE](#) backend
- +oauth - adds OAuth boilerplate for the [clj-oauth](#) library
- +cucumber - a profile for cucumber with `clj-webdriver`
- +sassc - adds support for [SASS/SCSS](#) files using [SassC](#) command line compiler
- +war - adds support of building WAR archives for deployment to servers such as Apache Tomcat (should *NOT* be used for [Immutable apps running on WildFly](#))



+site - creates template for site using the specified database (H2 by default) and ClojureScript

+kibit - adds [Kibit](#) static code analyzer support

+servlet - adds middleware to support the servlet context for running inside Java application servers

To add a profile simply pass it as an argument after your application name, eg:

```
lein new luminus myapp +cljs
```

You can also mix multiple profiles when creating the application, eg:

```
lein new luminus myapp +cljs +swagger +postgres
```

Note that some profiles will override others where appropriate. For example, the +re-frame profile has precedence over +reagent, so

```
lein new luminus myapp +reagent +re-frame
```

is equivalent to

```
lein new luminus myapp +re-frame
```