

# Introduction

# Table of Contents

|                          |   |
|--------------------------|---|
| Features .....           | 1 |
| Stable .....             | 1 |
| Under construction ..... | 2 |
| Limitations .....        | 2 |
| Java 8 .....             | 2 |
| Maven .....              | 2 |
| Version .....            | 3 |
| Browsers .....           | 3 |
| Java DSL .....           | 3 |

This is a web administration user interface for the integration test framework Citrus [www.citrusframework.org](http://www.citrusframework.org) written in Angular2. Major functionality objectives are project and configuration management as well as test execution and reporting.

## Features

The Citrus administration UI is there to help you manage configuration, modules and tests in your project. The user interface is able to give you a hand when it comes to managing Citrus messaging components and other configuration items such as (endpoints, validation Matcher, functions, schema repositories and so on).

The UI is browser based and is able to open and read your Citrus project. The administration pages provide access to the test cases so you can view and execute them. Of course you will also be able to review the test results. The administration web UI is not there to eliminate your favorite IDE (IntelliJ, Eclipse or whatever)! The UI is a helping instrument for getting in touch with Citrus and its concepts and works side by side with your local Java IDE as well as other text editors of your choice.

Also the UI is helpful when executing the Citrus integration tests in different stages (test, acceptance, explorative) of your release process. There is not always a full capable development environment available for executing integration tests. You can run the Citrus administrative UI as Docker container or Kubernetes pod in order to make the tests portable to your containerized test environment.

Now lets have a quick look at the feature set.

## Stable

Consider following features to be in a stable state:

- [Project](#)
  - [Open project](#)
  - [New project](#)
  - [Dashboard](#)
  - [Settings](#)
- [Configuration](#)
  - [Endpoints](#)
  - [Global Variables](#)
  - [Namespace Context](#)
  - [Data Dictionaries](#)
  - [Functions](#)

- [Validation Matcher](#)
- [Schema Repositories](#)
- [Tests](#)
  - [Search tests](#)
  - [List tests](#)
  - [Editor](#)
  - [Execute tests](#)

## Under construction

Consider following features to be under construction:

- [Project](#)
  - [Statistics](#)
- [Tests](#)
  - [Test designer](#)
  - [Reporting](#)

Please let us know if you are missing a feature and/or like to vote for features.

## Limitations

**NOTE: This project is still still under construction!**

The administration UI is stable not yet finished. Some features are still under construction. Some aspects are simply not covered yet.

Following from that we have to deal with some limitations and trade offs until the project emerges (hopefully with the help of the community, keeping our fingers crossed!). However the administration UI is usable and most features are considered to be stable.

Please consider following limitations that we have right now:

## Java 8

The administration UI is implemented using Java 8. You need at least Java 8 runtime in order to run the Citrus admin web application on your machine.

## Maven

At the moment we are limited to Citrus projects using Maven as build tool. Gradle projects are currently not supported. The build tool used is quite important as the administration UI is reading

the build information and uses Maven to execute test cases. We are working on Gradle support, please stay tuned and maybe poke us when things are not evolving.

## Version

Your project should use Citrus version 2.6 or higher in order to be able to work with the administration UI. Earlier Citrus versions might work too but are not tested and will not get support with bugfixes.

In addition to that we expect you to use the Citrus annotations when declaring test methods. The administration UI is looking for all tests in your project. It is required that you use **@CitrusTest** and **@CitrusXmlTest** annotations on your test methods. Otherwise the test cases will not be found and displayed.

## Browsers

At this early state we do not support browsers other than Chrome. This does not mean that other browsers are not working with the administration UI but the features are not tested yet with other browsers. The Citrus development team is using Chrome so you can be sure that errors related to browser incompatibility will be fixed very soon for Chrome.

## Java DSL

Citrus provides both XML and Java DSL for writing test cases. The administration UI is definitely able to read your XML test cases. The UI should also be able to read and manage your Java DSL test cases but there might be some limitations when reading your Java DSL code especially when you use object marshalling/unmarshalling as well as highly customized methods.

Please be curious and find out whether the current UI state is working with your project. And please let us know when there is something wrong.