SurePassID Swagger Codegen

SurePassID Authentication Server 2025

© 2013-2025 SurePassID, Corp. All rights reserved. Protected by patents pending. SurePassID, the SurePassID logo and design, and Secure SSO are registered trademarks or trademarks of SurePassID, Corp. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

SurePassID, Corp.

360 Central Avenue

First Central Tower

Suite 800

St. Petersburg, FL 33701

USA

+1 (888) 200-8144

www.surepassid.com

Introduction

The SurePassID FIDO2 and Fido UAF Server SDK is created using Swagger Codegen. The Swagger toolset provides a number of convenience tools for REST API development, the codegen component provides a means to quickly providing an SDK based on the Open API standard in a variety of languages. The languages and dependency management systems that SurePassID supports are listed below.

LANGUAGE	DEPENDENCY MANAGER
JAVA	Maven/Gradle
C#	Nuget
JAVASCRIPT	NPM
PHP	Composer

You can download the collection from:

https://sandbox.surepassid.com/downloads/SDK/swagger.zip

https://sandboxuaf.surepassid.com/spid_uafServer.xml

Swagger Codegen

Supported Language Usage

For languages that SurePassID supports, using the Swagger Codegen is quite simple. Developers will need to have and use a dependency manager. For Java development that would be Maven/Gradle and Nuget for C#.

The JavaScript version will work out of the box within Node.JS. A browser-based application will work but requires the use of something like browserify to bundle all of the required SDK code along with any application code that employs the SDK. Browserify's usage is detailed in a number of locations on the web, but particularly helpful examples can be found here, and here.

Other Languages

As Swagger Codegen provides the capability of quickly generating an SDK that works with the SurePassID FIDO server, developers are not inherently limited by the languages SurePassID has chosen to support. Using the Swagger Codegen CLI tool developers can quickly generate an SDK for numerous other languages not listed here. In order to use the Swagger Codegen CLI, simply clone the repository from GitHub. Once cloned, ensure that you have Java and Maven installed. With the repository cloned and both Java and Maven prerequisites installed as well, open a terminal and navigate to the directory in which the repository was cloned. Once there, type in the command "maven clean package". This command will build the Swagger Codegen CLI and enable running commands within the CLI. In order to

generate an SDK using the CLI you'll need to locate the file "swagger-codegen-cli.jar" within the repository, once it's been located, navigate to its location and issue the following command in the terminal.

```
java – jar ./swagger-codegen-cli.jar generate -i [swagger-json-filename] -l [programming-language] -o [output-directory-pathname]
```

The bracketed items such as "[swagger-json-filename]" represent values that you will need to insert into the command in order for it to run successfully, please note that these values should be inserted without brackets e.g. "C:\swagger-specifications.json" not "[C:\swagger-specifications.json]". When this has been run an SDK compliant with the chosen language will be generated and stored at the specified location.

For more info please see the full Codegen docs here:

https://swagger.io/tools/swagger-codegen/