SensioLabs

Sf Symfony

GETTING STARTED WITH SYMFONY 4

BLENDED LEARNING TRAINING PROGRAM

(face-to-face + e-learning)

Goals:

In two days only, you'll install the Symfony 4 framework and discover its well organized directory structure. You'll also learn how to build you very first MVC web pages with Twig, the Symfony's smart template engine, and how to process and validate forms. This training course will also give you some official best practices as well as tips and tricks to make your application fully internationalizable.

Duration: 2 days - 14 hours
Price: €1,290.00 / person
Code: SF4C1
Public: PHP web Developers
Requirements: Good knowledge of PHP and Object Oriented Programming

DISCOVERING THE SYMFONY 4 PROJECT

- General overview of the Symfony 4 project (source code, license, documentation, community, philosophy, ecosystem, etc.)
- Understanding the release cycle of new Symfony versions
- Bootstrapping a new Symfony 4 application project with Composer and Flex
- Taming project files and directory structure
- Installing third party modules (bundles, libraries) with Flex
- Finding help and documentation resources on the Internet
- Understanding the HTTP request / response processing workflow
- Discovering the concept of Symfony environments (development, production and test)
- Developing code according to Symfony coding and namming conventions
- Taming the CLI tool and its numerous useful commands

LEARNING HOW TO DEBUG CODE

- Installing the « Web Profiler » with Symfony Flex
- Enabling errors and incidents logging
- Displaying the « Web Debug Toolbar » on Web pages
- Browsing the « Web Profiler » debugging panels
- Analyzing exceptions stack traces and error messages
- Introspecting a variable contents

CONFIGURING THE APPLICATION

- Comparing the supported built-in configuration format: XML, YAML and PHP
- Configuring the application thanks to global parameters
- Configuration application's routes and URLs with annotations
- Overriding the configuration defined by a built-in or third-party « bundle »

GENERATING CONTENTS FOR THE END USER

- Introducing the Twig templates engine
- Comparing Twig and raw PHP templating strategies
- Accessing global variables from Twig templates (request, session, user)
- Using tags, filters and function to generate and format contents in a Twig template
- Making templates truly extensible thanks to the template inheritance feature
- Splitting templates into smaller templates chunks to avoid code duplications
- Rendering a Symfony controller from within a Twig template
- Generating URLs based on the application's routes configuration
- Rendering Symfony forms with custom Twig view helpers

DEVELOPPING CONTROLLER AND ANALYZING THE REQUEST

- Designing a controller class with its action methods
- Mapping URL patterns to Symfony controllers thanks to annotations
- Generating raw responses and Twig templates based responses
- Accessing client's information from the Request object
- Reading and writing session data
- Reading and writing cookies data
- Triggering a 404 error page when the resource is not found
- Triggering an internal redirect to another controller to avoid code duplications
- Redirecting the user to another page

INTERACTING WITH THE END USER THANKS TO FORMS

- Discovering the « Form » component
- Designing and processing simple web forms
- Adding validation constraints on submitted form fields' data
- Prototyping the form rendering thanks to Twig view helpers
- Gathering the form submitted data and processing them
- Sending the form validated and filtered data by e-mail to some recipient

INTERNATIONALIZING AND LOCALIZING THE USER INTERFACE

- Discovering the « Translation » component
- Changing the default user's locale
- Defining custom abstract translation keys in Twig templates
- Generating XLIFF translation catalogues skeletons
- Translating abstract translation keys in XLIFF catalogues
- Dealing with dynamic translations including pluralized sentences

- **PLATFORM E-LEARNING**