

Course code: **SF2**

Course title: **Web Development with Symfony2**

Days: 3

## Description:

### Course intended for:

The training has been developed for programmers, who would like to start working with the Symfony2 framework. It contains a complete set of information, starting from the basics to moderately advanced topics, which will be sufficient to develop a real Web application.

### Course objective:

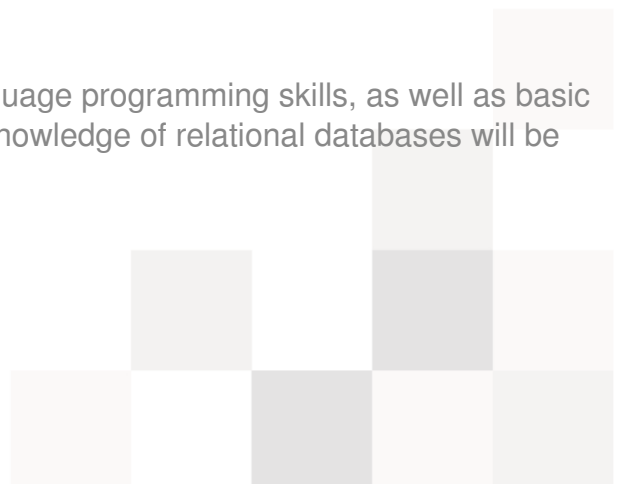
The training is aimed at presenting the practical applications of Symfony2 in Web application development. The trainers will discuss the essential issues associated with framework, supported by practical exercises. Step by step, the participants will build an exemplary application. In particular, the training participants will:

- Get familiar with the patterns and best practices used during work with Symfony2,
- Find out how to configure the application to suit their needs
- Learn to use the Twig template system,
- Get familiar with the aspects of validation of data entered by the user,
- Find out how to read and save data entities and ensure transactionability of database operations using the Doctrine library,
- Learn to develop their own services and work with an event-based code.

### Requirements:

The participants are expected to have the PHP language programming skills, as well as basic knowledge in object-oriented programming. Basic knowledge of relational databases will be an advantage.

### Course parameters:





3\*8 hours (3\*7 net hours) of lectures and workshops (with a visible emphasis on workshops)



## 1. Introduction

### I. General introduction to framework

#### i. The console tool

### II. Symphony2 Architecture

#### i. MVC, DI, DIC, IoC

#### ii. Handling of HTTP requests by the framework

### III. The project architecture

#### i. Structure and division into bundles

#### ii. External libraries

#### iii. Public catalog

### IV. Encoding conventions

#### i. PSR Standard

## 2. Project commencement

### I. Installation of Symphony2 in the standard version using the Composer tool

### II. Configuration of the application

#### i. Configuration formats

#### ii. Application environments

#### iii. Routing

### III. Web debug toolbar

### IV. Own bundle generation



3. Generating of responses for the client
  - I. Development of controllers
  - II. Session and cookie handling
  - III. Controlling of responses
    - i. Forwarding of responses
    - ii. Enforcing different response HTTP codes
  - IV. Generating of view templates overriding the controller
4. View templates using Twig
  - I. Introduction to Twig template engine
  - II. Layouts and inheriting of templates
  - III. Development of helpers:
    - i. Filters
    - ii. Functions
    - iii. Macros
  - IV. Supported template formats
5. Translations (i18n)
6. Forms and validation
  - I. Introduction to Form Framework
  - II. Development and handling of simple forms
  - III. Data validation basics
  - IV. Establishing of own validators
  - V. Rendering of forms using Twig templates
7. Database management using Doctrine



- I. Configuration of connection with the database
  - II. Repository and entity class development
  - III. Development and performance of custom queries
  - IV. Console commands for handling of the database structure
  - V. Event management using Lifecycle Callbacks
8. Security
- I. Introduction to the Firewall mechanism
  - II. User authentication
  - III. User authorization
  - IV. Introduction to FOSUserBundle (3rd party bundle)
9. REST Architecture
- I. Introduction to REST architecture
  - II. Building of own API
    - i. FOSRestBundle (3rd party bundle)
    - ii. NelmioApiDocBundle (3rd party bundle)
10. Container of services
- I. Introduction to the topic of services
  - II. Development of own services
  - III. Service tagging use
  - IV. Event handling using the EventDispatcher component
    - i. Listening to events
    - ii. Development of own events
    - iii. A review of bundles recommended by the community

