Formation PHP Symfony - 3

Sessions

Summary: *Following 42 formation course, you will learn about the Symfony security system, authentication and authorization.*

Version: 1

Contents

Chapter I

Foreword

Today we are going to learn about the Security component of the Symfony framework and how we can use it to create a register/login flow for our application.

   We will also learn how we can secure certain parts of out website for logged in users and even for users with certain roles.

Chapter II

General rules

- Your project must be realized in a virtual machine.

- Your virtual machine must have all the necessary software to complete your project. These softwares must be configured and installed.

- You can choose the operating system to use for your virtual machine.

- You must be able to use your virtual machine from a cluster computer.

- You must use a shared folder between your virtual machine and your host machine.

- During your evaluations you will use this folder to share with your repository.

- Your functions should not quit unexpectedly (segmentation fault, bus error, double free, etc) apart from undefined behaviors. If this happens, your project will be considered non functional and will receive a 0 during the evaluation.

- We encourage you to create test programs for your project even though this work **won't have to be submitted and won't be graded**. It will give you a chance to easily test your work and your peers' work. You will find those tests especially useful during your defence. Indeed, during defence, you are free to use your tests and/or the tests of the peer you are evaluating.

- Submit your work to your assigned git repository. Only the work in the git repository will be graded. If Deepthought is assigned to grade your work, it will be done after your peer-evaluations. If an error happens in any section of your work during Deepthought's grading, the evaluation will stop.

Chapter III

Day-specific rules

- For this day, your repository must contain just one working Symfony application.

- Best practices of the Symfony framework should be respected.

- Other third party bundles or libraries CAN be used.

- Each exercise will have its own bundle name ExxBundle, where xx is the number of the exercise.

- Each exercise will have its own base route in the form of /eXX, where XX is the number of the exercise.The controllers and actions can however have any name you want.

If no other explicit information is displayed, you must assume the following versions of languages :

- `PHP - Symfony LTS`

- `HTML 5`

- `CSS 3`

Chapter IV

Exercise 01

	Exercise
	Exercise 01: Login & Register
Turn-in directory : *ex/*	
Files to turn in : `Files and folders from your application`	
Allowed functions : `All methods`	

First set up a new Symfony application.Then create a bundle called E01Bundle.All of the requirements of this exercise will be put in the bundle.The following exercises will also be resolved in their respective bundles named accordingly.

Then set up the Symfony security and add a login and a register form.Your users will have to be loaded from a database.Then also add a homepage which should show a welcome message with the name of the currently logged in user and a logout button or a default welcome message and links to the login and register forms if the user is not logged in.

Chapter V

Exercise 02

	Exercise
Exercise 02: User Roles & Administration	
Turn-in directory : *ex/*	
Files to turn in : `Files and folders from your application`	
Allowed functions : `All methods`	

Create an Administrator role for your users.Administrators will have access to a page where all the registered users will be shown.

On this page, the administrator has the ability to delete a user.He should not be able to delete his own user.

Chapter VI

Exercise 03

	Exercise
	Exercise 03: Posts

Turn-in directory : *ex/*
Files to turn in : `Files and folders from your application`
Allowed functions : `All methods`

Create a new **Post** entity which should have at least the fields: title, content, created and author.The post should be linked to the user who created it.

The homepage should now display a list with all the posts created ordered from newest to oldest. The list should shown the title of the post, the name of its author and the creation date.

Each title should have a link to view the posts details page who should only be accessible by logged in users.The homepage should also have a link for logged in users to a form with the ability to create a new post.

Chapter VII

Exercise 04

	Exercise
	Exercise 04: Anonymous Users
Turn-in directory : *ex/*	
Files to turn in : `Files and folders from your application`	
Allowed functions : `All methods`	

Anonymous user sessions should last only a minute.When an anonymous user accesses the website, he will be assigned a random name which comes from a list of animals you define, preceded by the text *anonymous*. Eg: Anonymous dog

The name will be displayed on the homepage.Also a message will be displayed showing the time in seconds since the last request was made.

Chapter VIII

Exercise 05

	Exercise
	Exercise 05: Votes
Turn-in directory : *ex/*	
Files to turn in : `Files and folders from your application`	
Allowed functions : `All methods`	

Add ability for logged in users to vote on a post by liking or disliking it.Links for doing this can be found on the post detail page.A user can vote on a post only once.

The posts lists on the homepage should also display the amount of likes/dislikes a post has.

Each time a username appears on the website, besides it a *reputation* score should be shown. The reputation of a user is the sum of all the likes minus the sum of all the dislikes his own posts have received.

Chapter IX

Exercise 06

	Exercise
	Exercise 06: Post Editing
Turn-in directory : *ex/*	
Files to turn in : `Files and folders from your application`	
Allowed functions : `All methods`	

Logged in users should be able to edit a post.On the detail page of a post an edit button should be shown.

At the bottom of a post's detail page a text should appear saying which user last edited the post and at what date/time, if the post was edited.

Chapter X

Exercise 07

	Exercise
	Exercise 07: User Privileges
Turn-in directory : *ex/*	
Files to turn in : `Files and folders from your application`	
Allowed functions : `All methods`	

Depending on the number of reputation points a user has, he will have the following privileges:

- 0 points (new user) - right to create a post and edit their own post only

- 3 points - right to like posts

- 6 points - right to dislike posts

- 9+ points - right to edit any post

An admin has the right to do everything on the site regardless of his reputation points.

Also use Doctrine Fixtures to populate the database with test users, posts and votes to facilitate the testing of your application.

Chapter XI

Submission and peer-evaluation

Turn in your assignment in your `Git` repository as usual. Only the work inside your repository will be evaluated during the defense. Don't hesitate to double check the names of your folders and files to ensure they are correct.

> ℹ️ The evaluation process will happen on the computer of the evaluated group.