Back-End v15.0

↑ Back to 'Code Reviews Back-End'

Code Review 13

Good Morning Students!

Welcome to your last Code Review in this course.

You will need to achieve more than 60 points to pass this CodeReview successfully.

You will be graded according to the results achieved. Please see the number of points for each task below.

The grading system has the following rules:

Up to 60 points = red

Between 61 and 79 = yellow

Over 80 = green

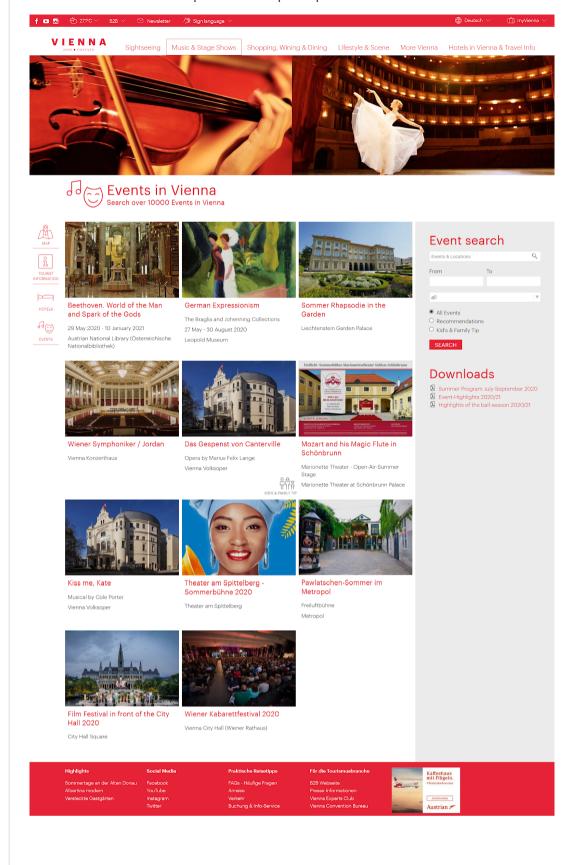You can submit your solution (as GitHub project link) until Saturday at 18:00:00.

Project description:

A global event management company needs a web application to track events. They would need to have a list of all their events with the following data:

- Event's name

- Event's date and start time

- Event's description

- Event's image (link to the image)

- Event's capacity (number of people)

- Event's contact e-mail

- Event's contact phone number

- Event's address (physical location, including street name and number, ZIP code and city name)

- Event's URL

- Event's Type (predefined list like "music", "sport", "movie", "theater" etc.). Note: This is related to the bonus points, not necessary for the basic grading.

NOTE: You can use as inspiration and perhaps also as a source of the content the WienTourismus official webpage: https://events.wien.info/en/

Project Naming:

Create a GitHub Repository named: BE15_CR13_YourName. Push the files into it and send the link through the learning management system (LMS). Please make your repository private and collaborate with codefactorygit. See an example of a GitHub link below:

https://github.com/JohnDoe/repositoryname.git.

Your MySQL database should be named: BE15_CR13_bigEventsYourName

You will need to implement this project using Symfony Version 6!

For this CodeReview, the following criteria will be graded:

- (20) Create a nice looking responsive design. Feel free to use Bootstrap or just HTML/CSS.

Implement an interface for the CRUD on Events:

- (20) Event index page: all events should be listed here like in the image above (event name, event date and time). Feel free to use Bootstrap cards to display the events.

- (20) Event view page (details page): when a user clicks on the event card/button, an event view page with all the data from that specific event should be displayed.

- (15) Event edit page: on this page, it should be possible to edit the event data.

- (15) Event add/create page: on this page, it should be possible to add a new event.

- (10) Event delete option: a user should be able to delete an event from the database.

Please note that the file upload is not mandatory for this Code Review.

Bonus Points:

- (20) Create filtering depending on the event type (hint: pass the information to the URL)

Hint: you could use the method findBy()

$repository = $this->getDoctrine()->getRepository(Event::class);

$events = $repository->findBy(['type' => 'music']);

This method takes an array as an argument and will return an array with all the results matching the search criteria.

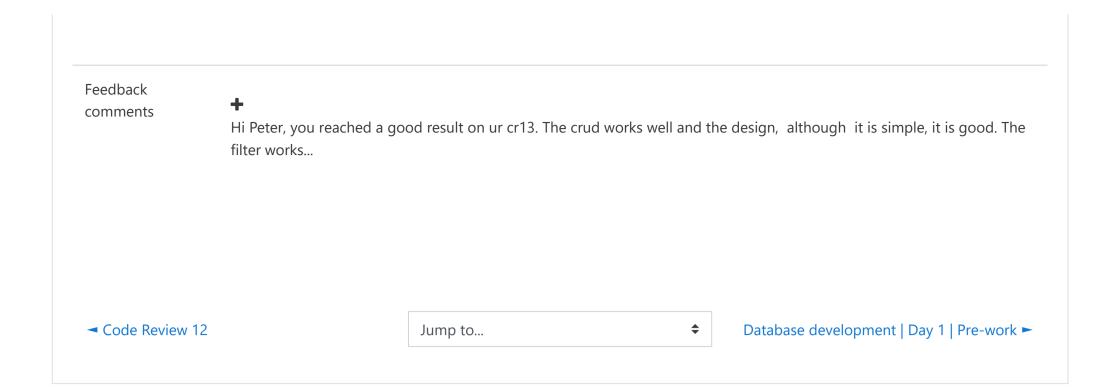**Note: Don't forget to upload your database to the GitHub repository.**

Submission status

Submission status	Submitted for grading
Grading status	Graded
Due date	Saturday, 9 April 2022, 6:00 PM
Time remaining	Assignment was submitted 8 days 5 hours late
Last modified	Sunday, 17 April 2022, 11:33 PM
Online text	➕ so, the challange was done: https://github.com/peru3232/BE15_CR13_Peter/invitations :-) happy easter!
Submission comments	➕ Comments (0)

Edit submission

You can still make changes to your submission

Feedback

Grade	120.00 / 120.00
Graded on	Tuesday, 19 April 2022, 3:21 PM
Graded by	Acilio Carraro

Feedback comments	**+**
	Hi Peter, you reached a good result on ur cr13. The crud works well and the design, although it is simple, it is good. The filter works...