



ข้อกำหนดทางเทคนิค

การแข่งขันฝีมือแรงงานแห่งชาติ ครั้งที่ 28

สาขา : เว็บไซต์

วันที่ 18 - 19 มีนาคม 2563

สถาบันพัฒนาฝีมือแรงงาน 1 สมุทรปราการ
อำเภอบางเสาธง จังหวัดสมุทรปราการ

ข้อกำหนดทางเทคนิคการแข่งขันฝีมือแรงงานแห่งชาติ ครั้งที่ 28 สาขาเว็บไซต์ดีไซน์

1. ลักษณะข้อสอบที่ใช้ในการแข่งขัน

ปฏิบัติงานด้านการสร้างเว็บแอปพลิเคชัน โดยจะต้องใช้ความรู้และทักษะในเรื่องต่างๆ ได้แก่ PHP, HTML, CSS, JS (Framework), JavaScript, MySQL, JQuery เป็นต้น เพื่อให้สามารถแสดงผลตรงตามโจทย์ที่กำหนดไว้ รวมถึงวัดทักษะการแก้ปัญหาตามที่ได้รับมอบหมายได้อย่างถูกต้อง

2. เวลาที่ใช้ในการแข่งขัน

จำนวน 9 ชั่วโมง

3. ข้อเสนอแนะ

- 3.1 ห้ามใช้หนังสืออ้างอิง จดบันทึก หรืออุปกรณ์เก็บข้อมูลนอกเหนือจากที่กำหนด
- 3.2 คณะกรรมการจะเป็นผู้กำหนดที่นั่งให้แก่ผู้เข้าแข่งขันแต่ละคน โดยการจับฉลาก
- 3.3 จะมีการอธิบายกติกาและโจทย์ให้ผู้เข้าแข่งขันเข้าใจก่อนการแข่งขัน
- 3.4 ไฟล์ข้อมูลต่างๆ ที่จำเป็นในการแข่งขัน จะถูกจัดเตรียมบนเซิร์ฟเวอร์ โดยผู้เข้าแข่งขันจะต้องดาวน์โหลดมาเพื่อใช้งานเอง ประกอบด้วยจำนวนไฟล์ X โฟลเดอร์ (XX ไฟล์)
- 3.5 ภายหลังเสร็จสิ้นการแข่งขัน ผู้เข้าแข่งขันจะต้องส่งผลงานผ่านเครื่องเซิร์ฟเวอร์ที่จัดเตรียมไว้ โดยกรรมการจะตรวจผลงานจากบนเซิร์ฟเวอร์เท่านั้น

4. กฎ กติกาในการแข่งขัน

- 4.1 ผู้เข้าแข่งขันจะต้องเป็นผู้ที่ได้รับการคัดเลือกโดยสถาบันพัฒนาฝีมือแรงงาน ในบางสาขาที่มีผู้เข้าแข่งขันไม่ครบตามจำนวนที่กำหนดให้คณะกรรมการเทคนิค สาขาเว็บไซต์ดีไซน์ จะคัดเลือกตัวแทนจากหน่วยงานภาครัฐหรือเอกชนเพื่อเข้าร่วม
- 4.2 ผู้แข่งขันต้องแต่งกายให้เรียบร้อยเหมาะสมปลอดภัย
- 4.3 ผู้แข่งขันต้องปฏิบัติงานด้วยความปลอดภัยและใช้วัสดุอย่างประหยัด
- 4.4 ผู้แข่งขันจะต้องมารายงานตัวต่อกรรมการผู้ควบคุมการแข่งขันไม่น้อยกว่า 30 นาที ก่อนเริ่มการแข่งขันทุกวัน เพื่อรับทราบคำแนะนำ คำชี้แจง และข้อปฏิบัติในการแข่งขัน
- 4.5 การตัดสินของกรรมการถือเป็นเด็ดขาด
- 4.6 ผู้ได้รับรางวัลบางสาขาอาจมีโอกาสดำเนินการพิจารณาให้เข้าเก็บตัวฝึกซ้อมและรับการคัดเลือกเป็นตัวแทนเข้าร่วมการแข่งขันฝีมือแรงงานอาเซียนหรือนานาชาติ โดยคณะกรรมการจะเป็นผู้คัดเลือกจากผู้ได้รับรางวัลทั้งหมดอีกครั้ง ตามหลักเกณฑ์ที่กำหนดขึ้น

5. เครื่องมือ วัสดุ อุปกรณ์ที่ใช้ในการแข่งขัน

5.1 วัสดุและอุปกรณ์ที่จัดเตรียมไว้ให้สำหรับผู้แข่งขันต่อหนึ่งคน

ลำดับ	รายการ	คุณลักษณะ	จำนวน	หน่วยนับ
1	เครื่องคอมพิวเตอร์		1	เครื่อง
2	Software	Software Workstation - Windows 10 (64 bit) - PHPStorm - WebStorm - Sublime (No Extension) - Atom (No Extension) - VSCode (No Extension) - Node 12.13 - PHP 7.3 + phpunit - Photoshop + Dreamweaver 2019 - MySQL Workbench - FileZilla Client - Chrome - Postman Framework JS - VueJs 2 (https://github.com/vuejs-templates/webpack) - Angular 8 - React 16.12 (npx create-react-app my-app) PHP - Laravel 6		
3	จอคอมพิวเตอร์		1	จอ
4	คีย์บอร์ด		1	อัน
5	เมาส์		1	อัน

5.2 วัสดุและอุปกรณ์ส่วนกลาง

ลำดับ	รายการ	คุณลักษณะ	จำนวน	หน่วยนับ
1	เครื่องเซิร์ฟเวอร์	Software Server - CentOS 7 - ESXi - PHP 7.3 + phpunit - Apache 2 - Node 12.13 - MariaDB 10.4 - PhpMyadmin 4.9 - SSH - FTP Server	1	เครื่อง
2	สวิตช์ layer		1	เครื่อง
3	ระบบเครือข่าย		1	ระบบ

5.3 วัสดุและอุปกรณ์ที่ผู้เข้าแข่งขันต้องเตรียมมา (หากไม่มีให้ระบุโดยใช้ข้อความว่า -ไม่มี- ให้ชัดเจน)
- ไม่มี -

ห้ามนำวัสดุ อุปกรณ์ใด ๆ เข้ามาใช้ในการแข่งขัน ห้ามนำอุปกรณ์สื่อสารและอุปกรณ์จัดเก็บข้อมูลใด ๆ เข้าไปในพื้นที่แข่งขัน

6. เกณฑ์การประเมิน

หัวข้อ	รายการ	คะแนน		
		Judgement (ถ้ามี)	Measurement	รวม
1	PHP and JavaScript	23.6	41	64.6
2	Speed Test	-	35.4	35.4
Total =		23.6	76.4	100

7. รายละเอียดในการให้คะแนน

7.1 หัวข้อ PHP and JavaScript รายละเอียดในการให้คะแนนประกอบด้วย

7.1.1 Database	6.85	คะแนน
7.1.2 Admin Backend	6.5	คะแนน
7.1.3 Ticket and Session	3	คะแนน
7.1.4 Organizer	6.85	คะแนน
7.1.5 REST API	7.85	คะแนน
7.1.6 Events list and agenda	7.5	คะแนน
7.1.7 Session and Events	6	คะแนน
7.1.8 General	8.35	คะแนน
7.1.9 JS Quality	5.35	คะแนน
7.1.10 Front end Design	6.35	คะแนน
รวม	64.6	คะแนน

7.2 หัวข้อ Speed Test รายละเอียดในการให้คะแนนประกอบด้วย

7.2.1 Speed Test	35.4	คะแนน
รวม	35.4	คะแนน



ใบสั่งงาน

การแข่งขันฝีมือแรงงานแห่งชาติ ครั้งที่ ๒๘

สาขาเว็บไซต์ดีไซน์

หัวข้อที่ ๑ PHP and JavaScript

ประกอบด้วย

๑. คำสั่งให้ผู้เข้าแข่งขันปฏิบัติงานสร้างเว็บไซต์ตามเงื่อนไขที่กำหนด
๒. ระยะเวลาการแข่งขัน ๖ ชั่วโมง (ระหว่างเวลา ๐๙.๐๐ – ๑๖.๐๐ น.)
๓. ข้อสอบที่ใช้ในการแข่งขันเป็นภาษาอังกฤษ ไม่มีการแปลเป็นภาษาไทย
๔. ผู้เข้าแข่งขันต้องส่งไฟล์ผลงานบนเซิร์ฟเวอร์ให้กรรมการตรวจให้คะแนน

โจทย์ในการปฏิบัติงาน

ตามเอกสาร Test Project หัวข้อ PHP and JavaScript

หัวข้อที่ ๒ Speed Test

ประกอบด้วย

๑. คำสั่งให้ผู้เข้าแข่งขันปฏิบัติงานสร้างเว็บไซต์ตามเงื่อนไขที่กำหนด
๒. ระยะเวลาการแข่งขัน ๓ ชั่วโมง (ระหว่างเวลา ๐๙.๐๐ – ๑๒.๐๐ น.)
๓. ข้อสอบที่ใช้ในการแข่งขันเป็นภาษาอังกฤษ ไม่มีการแปลเป็นภาษาไทย
๔. ผู้เข้าแข่งขันต้องส่งไฟล์ผลงานบนเซิร์ฟเวอร์ให้กรรมการตรวจให้คะแนน

โจทย์ในการปฏิบัติงาน

ตามเอกสาร Test Project หัวข้อ Speed Test

กำหนดการ

การแข่งขันฝีมือแรงงานแห่งชาติ ครั้งที่ 28

สาขาเว็บไซต์ดีไซน์

วันที่ 18 มีนาคม 2562

08.00 – 08.30 น.	ลงทะเบียนผู้เข้าแข่งขัน
08.30 – 09.00 น.	อธิบายข้อสอบ หัวข้อที่ 1 PHP and JavaScript
09.00 – 12.00 น.	แข่งขัน หัวข้อที่ 1 PHP and JavaScript
12.00 – 13.00 น.	พักรกลางวัน
13.00 – 16.00 น.	แข่งขัน หัวข้อที่ 1 PHP and JavaScript (ต่อ)

วันที่ 19 มีนาคม 2562

08.00 – 08.30 น.	ลงทะเบียนผู้เข้าแข่งขัน
08.30 – 09.00 น.	อธิบายข้อสอบ หัวข้อที่ 2 Speed Test
09.00 – 12.00 น.	แข่งขัน หัวข้อที่ 2 Speed Test
12.00 – 13.00 น.	พักรกลางวัน
13.00 – 16.00 น.	คณะกรรมการตรวจผลงานและรับรองผลการแข่งขัน



ใบสรุปคะแนน
การแข่งขันฝีมือแรงงานแห่งชาติ ครั้งที่ ๒๘
สาขาเว็บไซต์ดีไซน์

ลำดับที่	ชื่อ-สกุล ผู้เข้าแข่งขัน	คะแนนที่ได้	เหรียญรางวัล
๑			
๒			
๓			
๔			
๕			
๖			
๗			
๘			
๙			
๑๐			
๑๑			
๑๒			

หมายเหตุ เรียงรายชื่อตามลำดับคะแนนจากมากไปหาน้อย

ลงชื่อ..... ประธานอนุกรรมการฯ
(.....)

ลงชื่อ.....อนุกรรมการฯ
(.....)

ลงชื่อ.....อนุกรรมการฯ
(.....)

ลงชื่อ.....อนุกรรมการฯ
(.....)

ลงชื่อ.....อนุกรรมการฯ
(.....)

ลงชื่อ.....อนุกรรมการฯ
(.....)

ลงชื่อ.....อนุกรรมการฯ
(.....)

ลงชื่อ.....อนุกรรมการฯ
(.....)

ลงชื่อ.....อนุกรรมการฯ
(.....)

Test Project

Web Technologies

Speed Test

Submitted by:

Thomas Seng Hin Mak WS

Marcelo Strehl BR

Seunglyul Ryu KR

Abraham Quesada CR

Ilya Belyakov RU

Chorng-Shiuh Koong TW

Manuel Schaffner CH

Siew Hwa Kang SG

Zoltan Sisak HU

Contents

Contents	2
Introduction	3
Part A: Website Design	4
A1: Blur Picture Image (Easy).....	4
A2: Transparent Background (Easy).....	4
A3: Pattern (Easy)	4
A4: Loading GIF (Normal).....	4
A5: Walking Gif (Normal).....	4
Part B: Layout	5
B1: HTML tags (Easy)	5
B2: CSS Grid (Normal)	5
B3 Loading Screen (Normal)	6
B4 Toggle (Normal)	6
B5 Yellow Square (Easy).....	6
B6 Triangle (Easy).....	6
B7 Grayscale image (Easy)	6
B8 Jumper Ball (Easy).....	6
B9: HTML/CSS – Contact Form (Normal)	7
B10: Border Radius & Box Shadow (Easy).....	7
B11: Race track (Normal).....	7
B12: Video player interface (Normal)	7
B13: Gradient animation (Easy)	7
B14: Modern layout (Easy)	7
Part C: Front-End Development	9
C1: Loading Animation (Normal)	9
C2: Drawing Canvas (Normal)	9
C3: RGB Slider (Normal).....	9
C4: JS – Calculation (Easy).....	10
C5: List of expenses: (Easy)	10
C6: Canvas Animation (Easy)	10
C7: Smiley (Normal)	11
C8: Array Integers (Easy).....	11
C9: Following cursor (Easy).....	11
Part D: Back-End Development	12
D1: PHP Array (Easy)	12
D2: File manager (Normal).....	12
D3: A Pagination Code (Normal)	12
D4: XML2JSON Converter (Normal)	13
D5: Remove duplicates (Easy).....	13
Marking Scheme	14

Introduction

In this module, you are given 3 hours to complete mini tasks. There are 5 parts of mini tasks, they are design, layout, front-end development, back-end development and content management system.

You do not need to do all of the mini tasks. You will received instruction on which tasks that is required during competition.

Part A: Website Design

A1: Blur Picture Image (Easy)

Given a picture image, please use graphical software to make this on greyscale, put a blur effect, and save as a1.jpg.

A2: Transparent Background (Easy)

Given an image on format jpg. Please use graphical software to turn the background into transparent, and save as a2.png.

A3: Pattern (Easy)

Make a background image with any line pattern that repeat every 100x100 pixels, and save with 1920x1080 px as result.jpg.

A4: Loading GIF (Normal)

Create a GIF animation with a word "Loading" that appears on the left edge of image and moves outside of the right edge of image. The animation should be within 3 seconds to complete.

A5: Walking Gif (Normal)

Create a GIF animation from the given walk.gif file, so that the character moves in from the left edge of image and moves right and moves out of the right edge of screen.

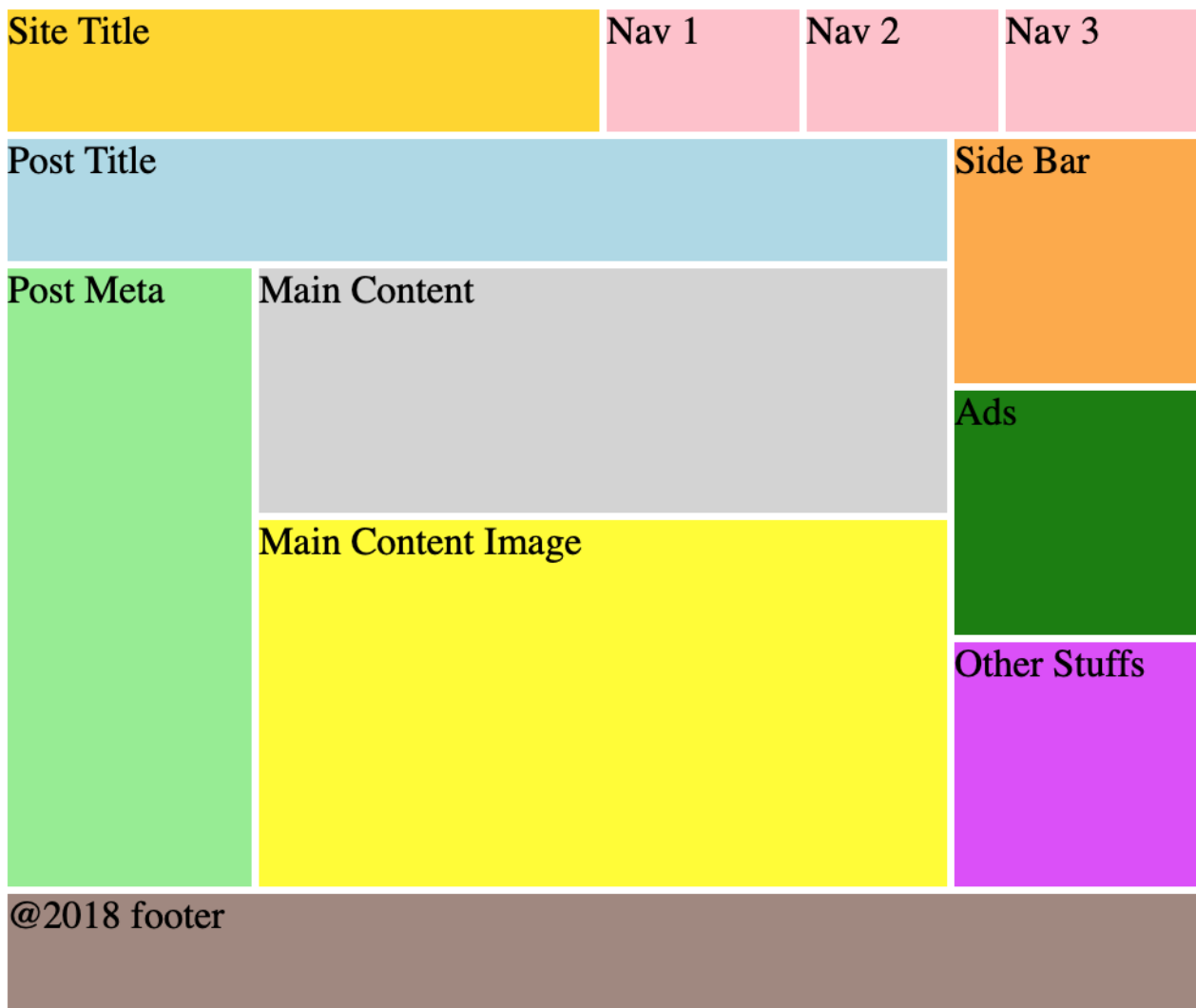
Part B: Layout

B1: HTML tags (Easy)

A customer wants you to make his writings into HTML so that it displays well in web browser. He has written the text. Your job is to use HTML to define the content for web browser to properly display it. Please consider semantic tags and accessibility.

B2: CSS Grid (Normal)

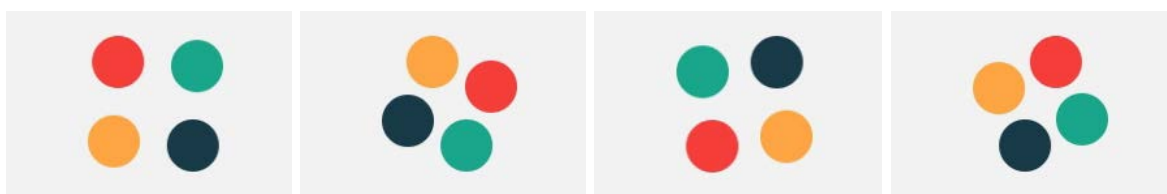
Please implement the following layout in CSS grid.



B3 Loading Screen (Normal)

Create an animated loader using only HTML and CSS same as video.mp4. The loader animation should be looped. 4 different colors (#19A68C, #F63D3A, #FDA543, #193B48) should be used.

Screen shots:



B4 Toggle (Normal)

To create an IOS Toggle switch using HTML and CSS only. View video for animation details.



B5 Yellow Square (Easy)

Make an animation that starts with a big yellow square, and this square turns a small red circle, the animation needs to be centered on the view and the animation needs to loop;

B6 Triangle (Easy)

Make a blue equilateral triangle on the center of the view, and animate this to transform a yellow square. the animation needs to be fluid and loop infinite;

B7 Grayscale image (Easy)

- Turn the image on grayscale, and when hover the mouse, the image need to be color with a transition;

B8 Jumper Ball (Easy)

- Make a loop animation of a jumper ball respecting all states from a animation (only css);

B9: HTML/CSS – Contact Form (Normal)

An HTML and CSS implementation of a not yet finished form are given.

Modify the CSS and HTML so that the form looks as depicted in the screenshot in the media files.

It's responsive behaviour should follow the screencast in the media files.

The use of JavaScript is NOT allowed.

B10: Border Radius & Box Shadow (Easy)

Create a page that contains a 400px rectangle in the middle.

Inside the rectangle there is a settings panel with inputs, where you can setup the rectangle's Border radius, and box shadow properties. The rectangle's css updates on input changes.

B11: Race track (Normal)

By using HTML and CSS, create a race track animation in the form of "8" (vertically or horizontally), along which the car will endlessly animated.

B12: Video player interface (Normal)

Create video player interface: search form, browse screen, action buttons (play, mute, and volume), progress slider, playlist (one item from playlist must be active), information about the video (title, description, number of views), comments (not less than 5) (and 1 attached comment).

B13: Gradient animation (Easy)

By using HTML and CSS, create a gradient shift animation on the page. The gradient may move into any direction.

B14: Modern layout (Easy)

Your task is to create the following layout, the posts' width is $\frac{1}{3}$ of the container's on desktop screen size ($\geq 1440px$) and full width below this.

Blog posts

Hello title

Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer.

[Read more](#)

Hello title

Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer.

[Read more](#)

Hello title

Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer.

[Read more](#)

Hello title

Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer.

[Read more](#)

Hello title

Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer.

[Read more](#)

Hello title

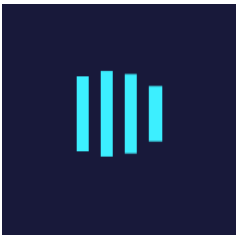
Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer. Lorem ipsum dolor sit amet. Libra dolo fabra milo, amer.

[Read more](#)

Part C: Front-End Development

C1: Loading Animation (Normal)

We want to create the following CSS loading animation.



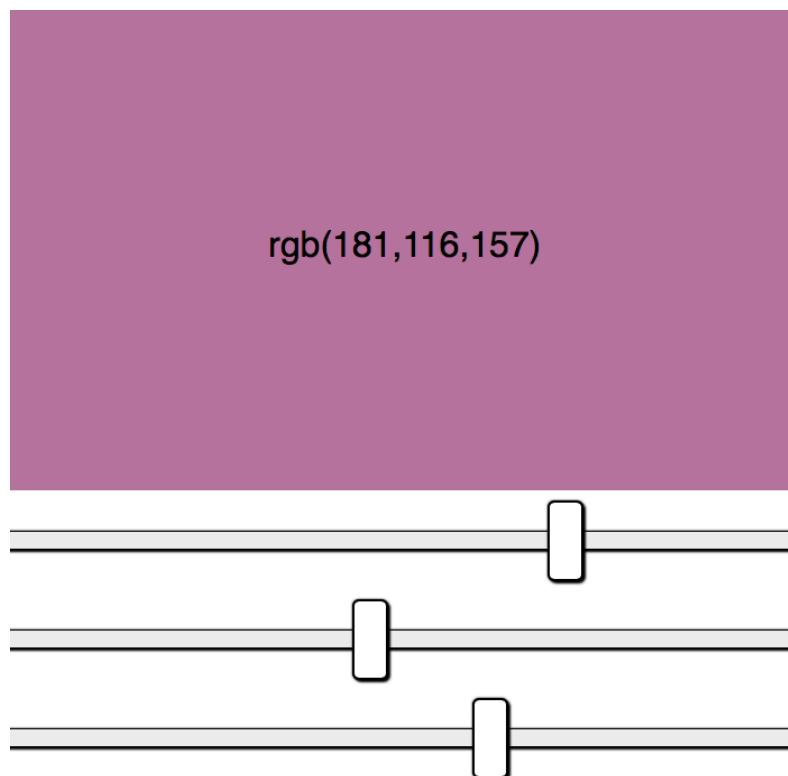
C2: Drawing Canvas (Normal)

Please try to use canvas to build a drawing pad with the following features:

- Basic drawing with mouse down and move
- 3 color switching

C3: RGB Slider (Normal)

Please implement an RGB slider as following. There are 3 sliders for adjusting values of red, green and blue.



C4: JS – Calculation (Easy)

Implement the function that calculate the sum of the squares of the odd numbers in a given array.

C5: List of expenses: (Easy)

a. Create a form with two fields called: **Spending name** and **Cost** and an **add** button.

b. Using JavaScript, perform the following functions:

1. Get the values of the fields.
2. Validate the entry of the data, if the fields are empty, print the message "There was an error", otherwise print "Correct".
3. Remove the expanded message and the data entered in the form after 3 seconds.
4. The data after being entered, must be shown in an ordered list using the HTML tags.

* No resources are provided.

C6: Canvas Animation (Easy)

Draw a circular figure in a container of 320px height and 400px width using canvas. When the browser window reloads, the circular object must move from left to right continuously.



C7: Smiley (Normal)

Create a big smiley face on the center of the page. It is 400px wide, and rectangular. Below the smiley, there are 3 buttons with texts: “:)””, “:|””, and “:(““. By clicking the buttons the smiley face changes to that state with a 0.5s animation. By default the smiley is in “:|”” state. The states have these designs:



C8: Array Integers (Easy)

Given an array that contains integers, rearrange on crescent order only in the positives integers, keeping the negatives on the same position (consoles the new array rearranged);

C9: Following cursor (Easy)

You have a div on the screen. make the div follows and be centered of accord a cursor position in the window;

Part D: Back-End Development

D1: PHP Array (Easy)

Given the code with some PHP arrays in media files, please implement a PHP function that compare the given two arrays and return a new array that contains the common elements from both given arrays.

D2: File manager (Normal)

Create a file manager to view the file system, delete files and folders, edit files and save them.



CROP

D3: A Pagination Code (Normal)

After reading data.json, the data in the current page must be shown..

5 data must be shown in one page.

5 page links to move to another page must be shown in one page

Create buttons to move 5 pages at once, previous and next.

Page link of current page must be located in the center

Take a look at the provided screen shots and data.json.

You can only use HTML & CSS & PHP, but for reading json data and the pagination code must use only PHP. You can use bootstrap and JavaScript is not allowed.

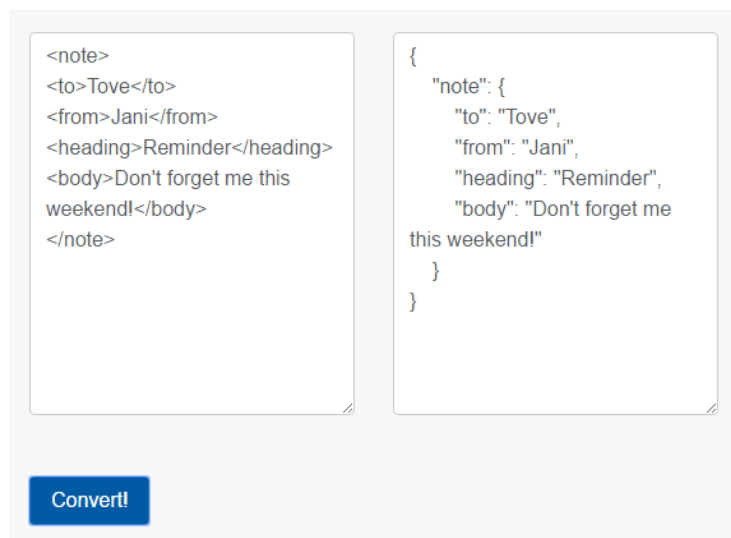
Screen shots:



D4: XML2JSON Converter (Normal)

Create a simple xml to json convertor using pure PHP(See the example.xml).

Screen shot:



D5: Remove duplicates (Easy)

Given an array, remove the duplicate values, and show the new array;

Marking Scheme

SECTION	CRITERION	JUDGEMENT MARKS	MEASUREMENT MARKS	TOTAL
A1	Design	0	5.2	5.2
A2	Layout	0	14.8	14.8
A3	Front-end	0	9.6	9.6
A4	Back-end	0	5.8	5.8

Test Project

Web Technologies

PHP and JavaScript Module

Submitted by:

Sebastian Häni – CH (Independent Test Project Designer)

Cyril Wanner – CH (Independent Test Project Designer)

Patrick Zahnd – CH (Independent Test Project Designer)

Contents

Contents	2
Introduction	3
Description of project and tasks	3
Phase one – PHP	4
Database design.....	4
Organizer interface	4
<i>Organizer login and logout</i>	5
<i>Event management</i>	6
<i>Ticket management</i>	9
<i>Session management</i>	10
<i>Channel management</i>	12
<i>Room management</i>	13
<i>Room capacity</i>	14
Attendee API.....	15
<i>Events overview</i>	15
<i>Event detail</i>	16
<i>Attendee login and logout</i>	17
<i>Event registration</i>	18
Phase two – JavaScript	19
API Setup	19
Introduction	20
Events list.....	21
Event agenda.....	22
Session detail.....	23
Event registration	24
Attendee login	26
Notes	27
Submission Instructions to the Competitor	28
Equipment, machinery, installations, and materials required	29
Marking Scheme	29

Introduction

The agency responsible for organizing the WorldSkills Conferences needs a new web application helping them to manage the events and allowing attendees to sign up and see all information they need. It should be dynamically built so that it can later be shared with other member organizations which means that multitenancy is an important requirement.

Because this application will grow over time, important requirements are covered by automated tests.

Description of project and tasks

The project is split into two phases:

- phase one (morning, 3 hrs) for building the backend parts (admin interface and REST-API) using PHP (framework) and database
- phase two (afternoon, 3 hrs) for building the frontend parts (attendee interface) using HTML/CSS/JS (framework) and the REST-API provided for phase two.

Phase one – PHP

Database design

- Use your database (dbYY_competitor_day2) and import the schema/dump provided
- Extend your database to be able to store (do not modify/extend any existing table except if it is specifically noted as they will be resetted before every test run)
 - Event ratings
 - Attendees should be able to comment and rate events they have attended
 - Rating should be a full number between 1 and 5, where 1 is the lowest and 5 the highest rating
 - The date and time of the rating should be registered
 - Session ratings
 - Additional to the event rating, attendees should be able to rate sessions they have attended
 - Rating should be a full number between 1 and 5, where 1 is the lowest and 5 the highest rating
 - The date and time of the rating should be registered
- Your additional tables need to follow the third normal form.

Organizer interface

Create an organizer interface (Server Side Rendered using a PHP Framework) using the templates provided to

- manage events, sessions, channels, rooms and tickets (create, read, update)
- create a bar chart to see the occupancy rates of all rooms
- create dynamic ticket prices
- support multiple tenants

Because different organizers will use this backend, keep security in mind. It should not be possible to access events of other organizers or update events of other organizers.

Organizer login and logout

Because registering as a new organizer is not within the scope of this project, please update the first two organizers in the provided dump so that it is possible to log in with the following credentials:

- Email: demo1@worldskills.org Password: demopass1
- Email: demo2@worldskills.org Password: demopass2

WorldSkills Event Platform

Please sign in

Email
Password

[Sign in](#)

Specification:

Feature: A1 – Log in and log out as an organizer: As an organizer, I would like to log in and log out to the event platform so that I can access the functionality of the platform.

Scenario: Ala – Organizer login ok

Given I am on the "login" page

When I enter the specified email and password for an organizer

And press the "Sign in" button

Then I am forwarded to the organizer area

And organizer name and "Sign out" button become visible in the header

And page "Manage events" becomes visible

Scenario: Alb – Organizer login fail

Given I am on the "login" page

When I enter a wrong email and password combination for an organizer
or use login information of an attendee instead of an organizer

And press the "Sign in" button

Then I stay on the "login" page

And I see an error message "Email or password not correct"

Scenario: Alc – Organizer logout

Given I am logged in

When I press the "Sign out" button

Then I am forwarded to the "login" page

And organizer name and logout button and admin functions disappear

And I cannot reach the previous page by going back

Event management

Organizers should be able to save all information about an event and its sessions in the database. Every organizer account can only see and edit its own events. The organizer may have zero or many events.

Specification:

Feature: A2 - Manage events: As an organizer I want to manage events so attendees can find events

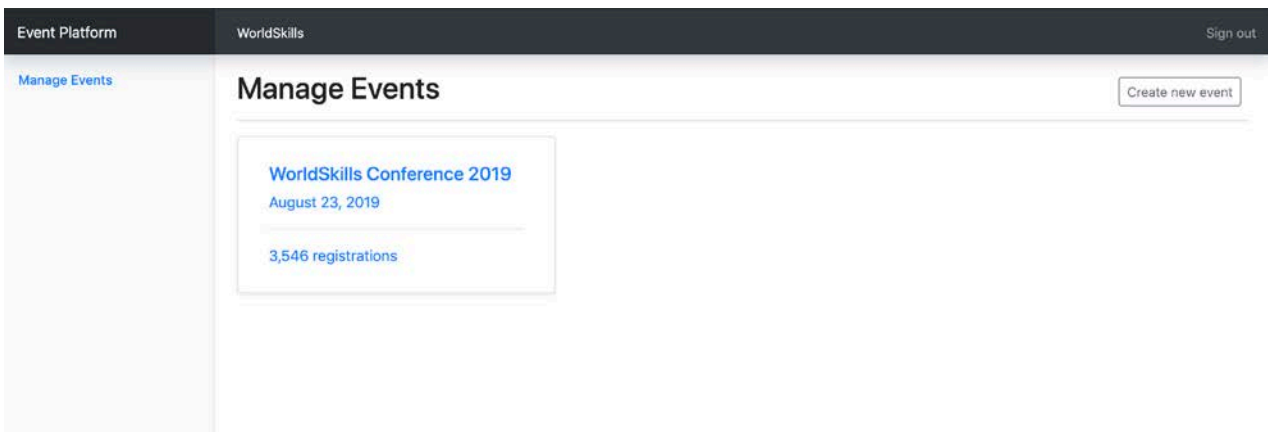
Scenario: A2a - List all events

Given I am logged in

And I am on the "Manage events" page

Then I see a list of all events created by the current organizer sorted ascendingly by their date

And I see the total number of registrations for every event



The screenshot shows the 'Manage Events' interface. At the top, there is a dark navigation bar with 'Event Platform' on the left, 'WorldSkills' in the center, and 'Sign out' on the right. Below the navigation bar, the page title 'Manage Events' is displayed in the center, with a 'Create new event' button to its right. A list of events is shown below, with the first event being 'WorldSkills Conference 2019' on August 23, 2019, with 3,546 registrations. The event details are enclosed in a light blue box.

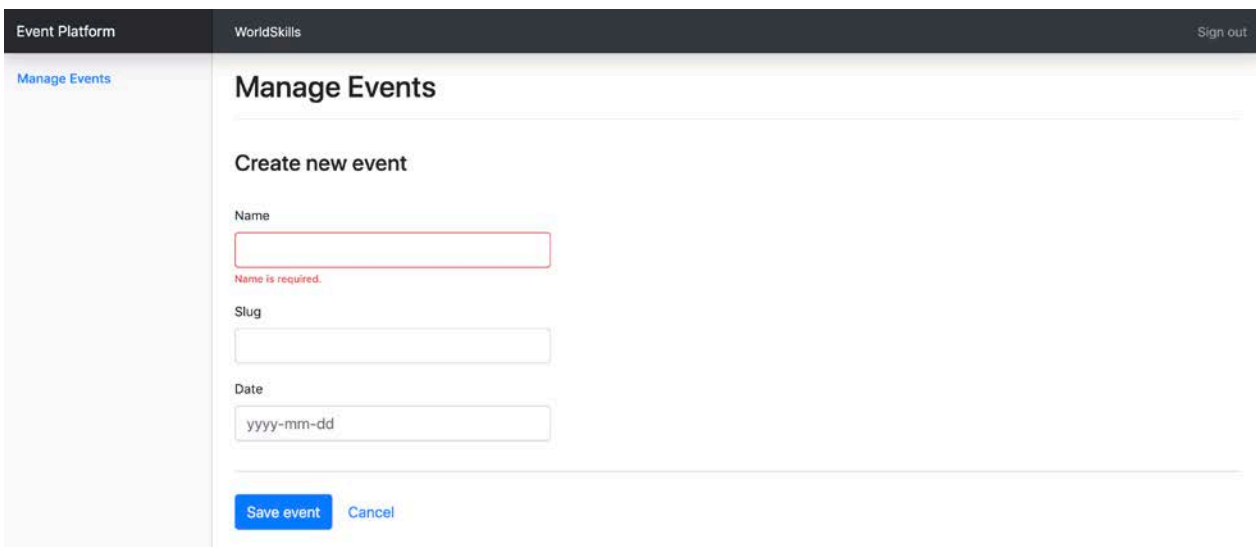
Scenario: A2b - Show form for new event

Given I am logged in

And I am on the "Manage events" page

When I press "Create new event"

Then a form to enter event information appears



The screenshot shows the 'Create new event' form. At the top, there is a dark navigation bar with 'Event Platform' on the left, 'WorldSkills' in the center, and 'Sign out' on the right. Below the navigation bar, the page title 'Manage Events' is displayed in the center. Underneath, the 'Create new event' form is shown. It has three input fields: 'Name' (with a red border and a 'Name is required.' error message below it), 'Slug', and 'Date' (with a placeholder 'yyyy-mm-dd'). At the bottom of the form, there are two buttons: 'Save event' (in blue) and 'Cancel'.

Scenario: A2c - Enter information for a new event

Given a form to enter event information is shown

When I enter info for the event (all fields required)

And I press "Save event"

Then the event is saved into the database

And I am redirected to the detail page for the newly created event

And a success message "Event successfully created" is shown

Scenario: A2d - Enter already used slug for a new event

Given a form to enter event information is shown

When I enter info for the event and use a slug which is already used by another event of the same organizer

And I press "Save event"

Then the event is not saved into the database

And an error message "Slug is already used" is shown

And my previously submitted input data is not lost

Scenario: A2e - Enter invalid slug for an event

Given a form to enter event information is shown

When I enter an invalid slug for the event

And I press "Save event"

Then the event is not saved into the database

And an error message "Slug must not be empty and only contain a-z, 0-9 and '-' is shown

Scenario: A2f - Show form for updating event

Given I am logged in

And I am on the detail page of an event

When I press "Edit event"

Then a form to update event information appears

Scenario: A2g - Update event information

Given a form to update information of an existing event

When I change the prefilled old event information to new values

And I press "Save event"

Then the updated event is saved into the database

And I am redirected to the detail page for the updated event

And a success message "Event successfully updated" is shown

Scenario: A2h - Show event detail

Given I am on the "Manage events" page

And I click on an event in the list

Then I am forwarded to the detail page on this event

And can see all basic information as well as all relations to the available rooms, channels, sessions and tickets

Event Platform
WorldSkills
Sign out

Manage Events

WORLDSKILLS CONFERENCE 2019

[Overview](#)

REPORTS

Room capacity

WorldSkills Conference 2019

August 23, 2019

[Edit event](#)

Tickets

[Create new ticket](#)

Normal

200.-

Early Bird

120.-

Available until June 1, 2019

VIP

400.-

100 tickets available

Sessions

[Create new session](#)

Time	Type	Title	Speaker	Channel
08:30 - 10:00	Talk	Keynote	An important person	Main / Room A
10:15 - 11:00	Talk	What's new in X?	Another person	Main / Room A
10:15 - 11:00	Workshop	Hands-on with Y	Another person	Side / Room C

Channels

[Create new channel](#)

Main

3 sessions, 1 room

Side

15 sessions, 2 rooms

Rooms

[Create new room](#)

Name	Capacity
Room A	1,000
Room B	100
Room C	100
Room D	250

Ticket management

For each event, organizers want to create different ticket types. Tickets can have, but don't need, one validity rule which can be one of:

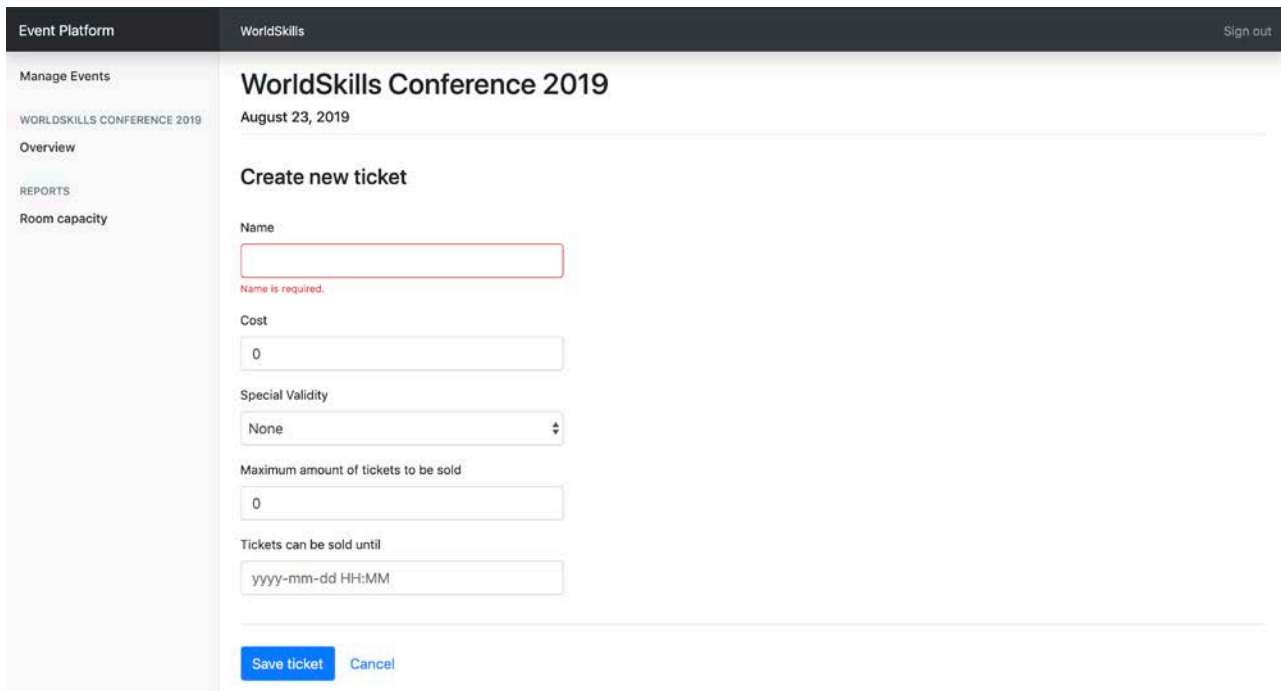
- only purchasable until a given date (e.g. book before July 30, 2019 for an early bird ticket)
- only a defined amount of tickets is available (e.g. max 50 VIP tickets)

Please check existing records in the provided dump to see how validity rules should be stored.

Specification:

Feature: A3 - Manage tickets: As an organizer I want to manage tickets that attendees can buy in the frontend

Scenario: A3a - Show form for new ticket
Given I am on the detail page of an event
And I press "Create new ticket"
Then a form to enter ticket information appears



The screenshot shows a web interface for managing events. The main content area is titled "WorldSkills Conference 2019" with the date "August 23, 2019". Below this, there is a section titled "Create new ticket". The form contains several input fields: "Name" (with a red border and a "Name is required." error message), "Cost" (with "0" entered), "Special Validity" (a dropdown menu set to "None"), "Maximum amount of tickets to be sold" (with "0" entered), and "Tickets can be sold until" (with a placeholder "yyyy-mm-dd HH:MM"). At the bottom of the form are two buttons: "Save ticket" and "Cancel".

Scenario: A3b - Enter information for a new ticket
Given a form to enter session information is shown
When I enter information for the ticket (name and cost field are required, also any field that is related to the selected special validity rule)
And I press "Save ticket"
Then the ticket is saved into the database
And I am redirected to the detail page of the event it is assigned to
And a success message "Ticket successfully created" is shown

Session management

For every event, organizers want to be able to list the current sessions and have the possibility to create new ones. Sessions are always assigned to a room which can not have another session during that time. Sessions are one of these two types:

- talk: A speaker presents a talk. Talks are included in the conference ticket and do not cost extra.
- workshop: A speaker holds a hand-on workshop with its attendees. Workshops can cost extra.

Specification:

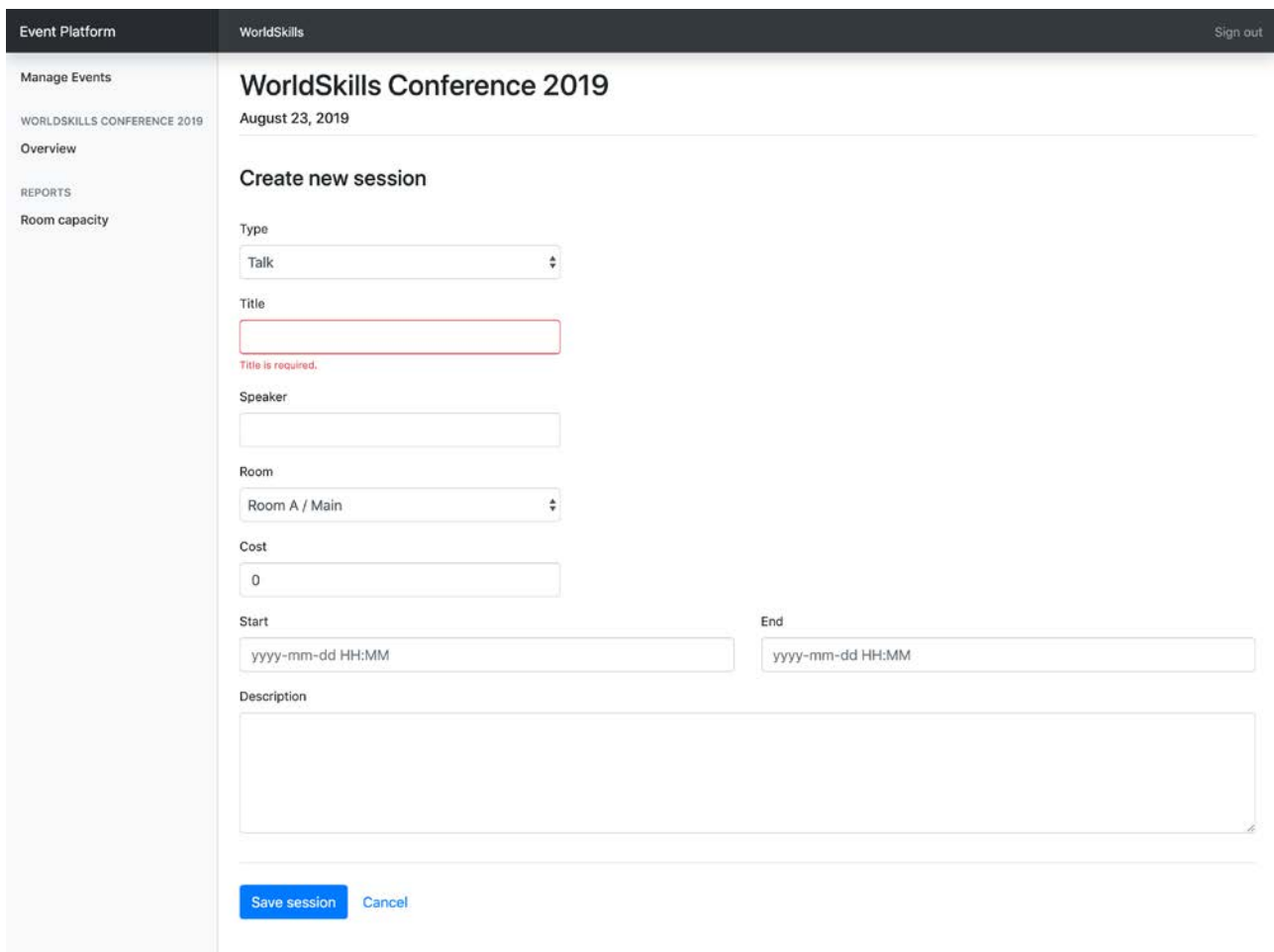
Feature: A4 – Manage sessions: As an organizer I want to manage sessions that attendees can sign up for

Scenario: A4a – Show form for new session

Given I am on the detail page of an event

And I press "Create new session"

Then a form to enter session information appears



The screenshot shows a web interface for managing sessions. The header includes 'Event Platform' and 'WorldSkills' with a 'Sign out' link. The main content area is titled 'WorldSkills Conference 2019' and 'August 23, 2019'. A sidebar on the left contains navigation links: 'Manage Events', 'WORLDSKILLS CONFERENCE 2019', 'Overview', 'REPORTS', and 'Room capacity'. The main form is titled 'Create new session' and contains the following fields:

- Type:** A dropdown menu with 'Talk' selected.
- Title:** A text input field with a red border and a red error message below it: 'Title is required.'
- Speaker:** A text input field.
- Room:** A dropdown menu with 'Room A / Main' selected.
- Cost:** A text input field with the value '0'.
- Start:** A date and time input field with the placeholder 'yyyy-mm-dd HH:MM'.
- End:** A date and time input field with the placeholder 'yyyy-mm-dd HH:MM'.
- Description:** A large text area for entering session details.

At the bottom of the form, there are two buttons: 'Save session' (in blue) and 'Cancel'.

Scenario: A4b - Enter information for a new session

Given a form to enter session information is shown

When I enter information for the session (all fields, except cost, required)

And I press "Save session"

Then the session is saved into the database

And I am redirected to the detail page of the event it is assigned to

And a success message "Session successfully created" is shown

Scenario: A4c - Enter invalid information for a new session - room booked

Given a form to enter session information is shown

When I choose a room which already has another session at the same time

And I press "save session"

Then the session is not saved into the database

And an error message "Room already booked during this time" is shown

And my previously submitted input data is not lost

Scenario: A4d - Show form for updating session

Given I am on the detail page of an event

And I click on the name of a session

Then a form to update session information appears

Scenario: A4e - Update session information

Given a form to update information of an existing session

When I change the prefilled old event information to new values

And I press "Save session"

Then the updated session is saved into the database

And I am redirected to the detail page of the event it is assigned to

And a success message "Session successfully updated" is shown

Channel management

A channel is used to group multiple sessions by a topic, for example management or user experience. Channels are assigned to exactly one event.

Specification:

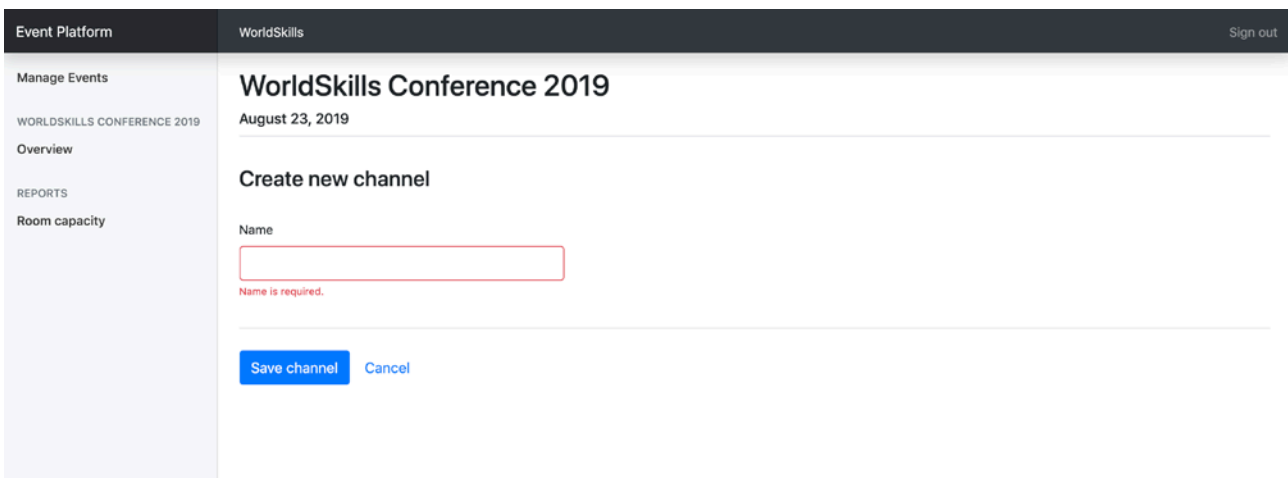
Feature: A5 - Manage channels: As an organizer I want to manage channels of an event so sessions can get assigned to them

Scenario: A5a - Show form for new channel

Given I am on the detail page of an event

And I press "Create new channel"

Then a form to enter channel information appears



The screenshot shows a web interface for the 'WorldSkills Conference 2019' event. The page title is 'WorldSkills Conference 2019' with the date 'August 23, 2019'. A sidebar on the left contains navigation links: 'Manage Events', 'WORLD SKILLS CONFERENCE 2019', 'Overview', 'REPORTS', and 'Room capacity'. The main content area is titled 'Create new channel' and features a form with a 'Name' field. The field is empty and has a red border, with a red error message 'Name is required.' below it. At the bottom of the form are two buttons: 'Save channel' (in blue) and 'Cancel' (in light blue). The top navigation bar includes 'Event Platform', 'WorldSkills', and a 'Sign out' link.

Scenario: A5b - Enter information for a new channel

Given a form to enter channel information is shown

When I enter information for the channel (all fields required)

And I press "Save channel"

Then the channel is saved into the database

And I am redirected to the detail page of the event it is assigned to

And a success message "Channel successfully created" is shown

Room management

A room is assigned to exactly one channel and can have exactly one session at a time.

Specification:

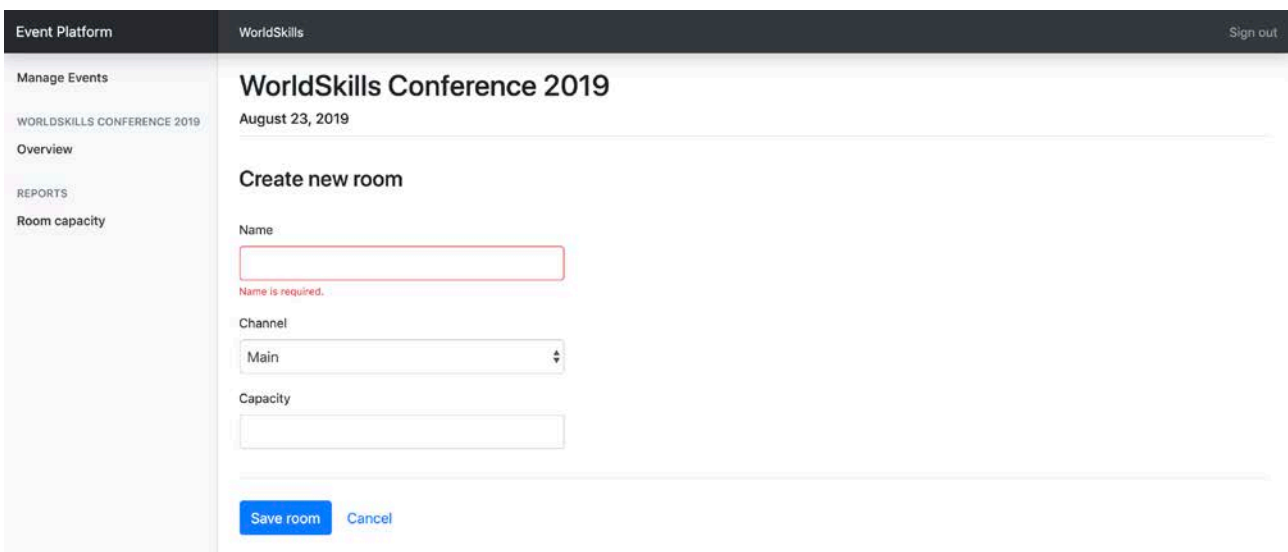
Feature: A6 - Manage rooms: As an organizer I want to manage rooms which I can later assign to sessions

Scenario: A6a - Show form for new room

Given I am on the detail page of an event

And I press "Create new room"

Then a form to enter room information appears



The screenshot shows a web interface for managing events. The top navigation bar includes 'Event Platform', 'WorldSkills', and a 'Sign out' link. The main content area is titled 'WorldSkills Conference 2019' and 'August 23, 2019'. A sidebar on the left contains navigation links: 'Manage Events', 'WORLD SKILLS CONFERENCE 2019', 'Overview', 'REPORTS', and 'Room capacity'. The main content area displays a 'Create new room' form with the following fields: 'Name' (a text input field with a red border and a 'Name is required.' error message below it), 'Channel' (a dropdown menu with 'Main' selected), and 'Capacity' (a text input field). At the bottom of the form are two buttons: 'Save room' (a blue button) and 'Cancel' (a light blue button).

Scenario: A6b - Enter information for a new room

Given a form to enter room information is shown

When I enter information for the room (all fields required)

And I press "Save room"

Then the room is saved into the database

And I am redirected to the detail page of the event it is assigned to

And a success message "Room successfully created" is shown

Room capacity

Organizers want to see an overview of all sessions with information about the room capacity and number of attendees. With this information, they can see if a room is reaching its full capacity.

Please note that for all sessions of type talk, attendees don't need a separate ticket. For sessions of type workshop, they need a separate ticket.

Specification:

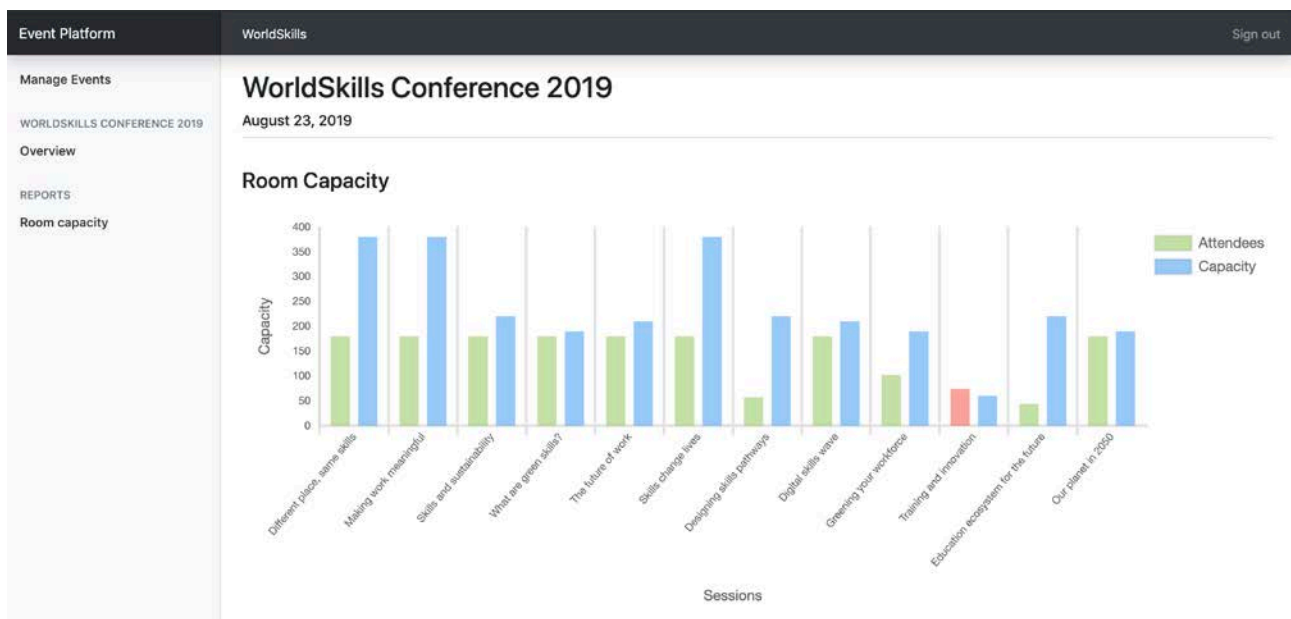
Feature: A7 - Room overview: As an organizer I want to see an overview of the sessions with the room capacity and number of attendees

Scenario: A7a - Show room overview

Given I am on the detail page of an event

When I press the link "Room capacity" in the sidebar

Then I see a bar chart displaying sessions along the horizontal axis, while the y-axis is showing the room capacity superimposed with the registered amount of attendees



Scenario: A7b - Room capacity exceeded

Given I am on the "Room capacity" page

When I have a room which exceeds its capacity

Then I see the visitors bar highlighted with a red color instead of green

Attendee API

Create a RESTful API (http://<hostname>/XX_JS-PHP_A/api/v1/...) to

- login and logout as an attendee
- read all upcoming events
- see the agenda for an event
- sign up and purchase a ticket for an event

PHPUnit tests

Tests for the PHP API are distributed with the mediafiles. Please see the following instructions to set them up and configure on your own server.

Setup:

1. Upload the API tests to your server
2. Configure the `.env` file of the API tests and provide correct database credentials and the URL where your API is available.

Running the tests:

Execute the following command on the server inside the directory of the api tests:

```
composer test
```

To only run a single test, you can use the following command for example:

```
composer test -- --filter B2EventDetailTest
```

Additional information for the REST-API tests/specification: the body contains some static example data, adjust it to fit the requirements. Placeholder parameters in the URL are marked with curly braces: {slug}

The tests are referenced by its function name [`-> test<scenario-id><text>`].

The order of properties in objects does not matter, but the order of items in an array does.

Events overview

When an attendee lands on the index page of the application, he will get a list with all upcoming events across all organizers. Events should be ordered ascendingly by their date.

Specification:

Feature: B1 – Read events: As a visitor I want to read the public data of all upcoming events so that I can display them in a list

Scenario: Bla – Read all upcoming events

Request

URL: /api/v1/events

Method: GET

Header:

Body: -

Response

If success [`-> testBlaGetEvents`]

Header: Response code: 200

```
Body: {"events": [{"id": 1, "name": "someText", "slug": "some-text",
"date": "2019-08-15", "organizer": {"id": 1, "name": "someText", "slug":
"some-text"}}]}
```

Event detail

When an event is selected, all information of the selected event should be returned in the same endpoint.

If a session has no cost or a ticket no description, NULL will be used.

The description of a ticket is generated as following:

- NULL if no special validity rule set
- "Available until {month} {day}, {year}" if special validity rule "date" is used
(example: "Available until September 1, 2019")
- "X tickets available" if special validity rule "amount" is used
(example: "30 tickets available" where 30 is the total amount of tickets and not the remaining ones)

Specification:

Feature: B2 - Read event: As a visitor I want to read all information about an event

Scenario: B2a - Read all detail information of a single event

Request

URL: /api/v1/organizers/{organizer-slug}/events/{event-slug}

Method: GET

Header:

Body: -

Response

If success [-> *testB2aGetEventDetail* | *testB2aTicketValidity*]

Header: Response code: 200

```
Body: {"id": 1, "name": "someText", "slug": "some-text", "date":
"2019-08-15", "channels": [{"id": 1, "name": "someText", "rooms": [{"id": 1,
"name": "someText", "sessions": [{"id": 1, "title": "someText",
"description": "someText", "speaker": "someText", "start": "2019-08-15
10:00:00", "end": "2019-08-15 10:45:00", "type": "workshop", "cost":
50|null}]}]}], "tickets": [{"id": 1, "name": "someText", "description":
"Available until July 7, 2019|null", "cost": 199.99, "available": true}]}
```

If event does not exist or was not created by the organizer [-> *testB2aGetInvalidEvent*]

Header: Response code: 404

Body: {"message": "Event not found"}

If organizer does not exist [-> *testB2aGetInvalidOrganizer*]

Header: Response code: 404

Body: {"message": "Organizer not found"}

Attendee login and logout

After signup (which is not within the scope of this project), attendees get assigned a random 6 characters long registration code which they need when entering an event. To avoid requiring yet another password, this registration code is also used for logging the user in on the website, together with his lastname as his login credentials. The registration code is shared across all events and is *not* only valid for a single event.

For testing purposes, the registration code and last name can be seen in the database dump.

On a successful login, data of the loggedin attendee gets returned together with a token which can be submitted with subsequent requests. For simplicity, the token is the username md5-hashed and is valid until the logout endpoint is called.

Specification:

Feature: B3 - Login and Logout as attendee: As a visitor I want to login/logout so that I can attend an event and see my registrations

Scenario: B3a - Attendee login

Request

URL: /api/v1/login

Method: POST

Header:

Body: {"lastname": "someText", "registration_code": "someText"}

Response

If success [-> *testB3aCorrectLogin/SameLastname/SameRegistrationCode*]

Header: Response code: 200

Body: {"firstname": "someText", "lastname": "someText", "username": "someText", "email": "someText", "token": "AUTHORIZATION_TOKEN"} (md5 hashed username, valid until logout)

If user login info not correct [->

testB3aInvalidLastname/RegistrationCode/Request]

Header: Response code: 401

Body: {"message": "Invalid login"}

Scenario: B3b - Attendee logout

Request

URL: /api/v1/logout?token={AUTHORIZATION_TOKEN}

Method: POST

Header:

Body: -

Response

If success [-> *testB3bLogout*]

Header: Response code: 200

Body: {"message": "Logout success"}

If invalid token [-> *testB3bInvalidToken | testB3bAlreadyLoggedOut*]

Header: Response code: 401

Body: {"message": "Invalid token"}

Event registration

Users will register and buy a ticket with this endpoint. The different ticket validity rules (date or amount) should be validated. Registrations should be sorted ascendingly by their id.

Specification:

Feature: B4 - Event registration: As a visitor I want to register for an event

Scenario: B4a - New event registration

Request

URL: /api/v1/organizers/{organizer-slug}/events/{event-slug}/registration?token={AUTHORIZATION_TOKEN}

Method: POST

Header:

Body: {"ticket_id": 1, "session_ids": [1, 2, 3]} (session_ids is optional)

Response

If success [-> *testB4aCorrectRegistration*]

Header: Response code: 200

Body: {"message": "Registration successful"}

If user is not logged in or token is invalid [-> *testB4aLoggedOut*]

Header: Response code: 401

Body: {"message": "User not logged in"}

If user already registered for this event [-> *testB4aAlreadyRegistered*]

Header: Response code: 401

Body: {"message": "User already registered"}

If ticket is not available anymore [-> *testB4aInvalidTicket*]

Header: Response code: 401

Body: {"message": "Ticket is no longer available"}

Scenario: B4b - Get registrations of a user

Request

URL: /api/v1/registrations?token={AUTHORIZATION_TOKEN}

Method: GET

Header:

Body: -

Response

If success [-> *testB4bGetRegistrations* | *testB4bNewRegistration*]

Header: Response code: 200

Body: {"registrations": [{"event": {"id": 1, "name": "someText", "slug": "some-text", "date": "2019-08-15", "organizer": {"id": 1, "name": "someText", "slug": "some-text"}}, {"session_ids": [1, 2, 3]}]}

If user is not logged in or token is invalid [-> *testB4bLoggedOut*]

Header: Response code: 401

Body: {"message": "User not logged in"}

Phase two – JavaScript

API Setup

The API needed for this part will be **provided after lunch** and can be set up with the following instructions. In case you want to start with part two before lunch, you have to use your own API from part one.

Automated setup:

Connect to your server through SSH and execute the following command (in your home directory):

```
setup-api
```

This will automatically set up the API and database on your server. After completion, the API is available at <http://XX.skill17.com/day2-afternoon/> (XX is your workstation number).

The API is implemented according to the specification of the Attendees API in part one.

Manual setup:

The manual setup should only be done in case the automatic setup fails.

1. Download the API from http://wsi.skill17.com/day2_afternoon_api.zip (only available in the afternoon)
2. Upload it to your server and unzip it into the folder `/home/competitor/www/day2-afternoon` (ensure that the unzipped files are directly located in the folder `/home/competitor/www/day2-afternoon` and not in another subfolder)
3. Create a new database called `day2_afternoon` and import the provided dump from part one
4. The API is now available at <http://XX.skill17.com/day2-afternoon/> (XX is your workstation number)

Introduction

Using the RESTful API (all data coming/going into database), create a frontend for attendees to browse information about the events, login/logout and register for an event. During registration, it should be possible to add additional workshops they want to attend.

The RESTful API supports POST requests with JSON or FormData bodies, but the correct request headers need to be set accordingly to the used body type.

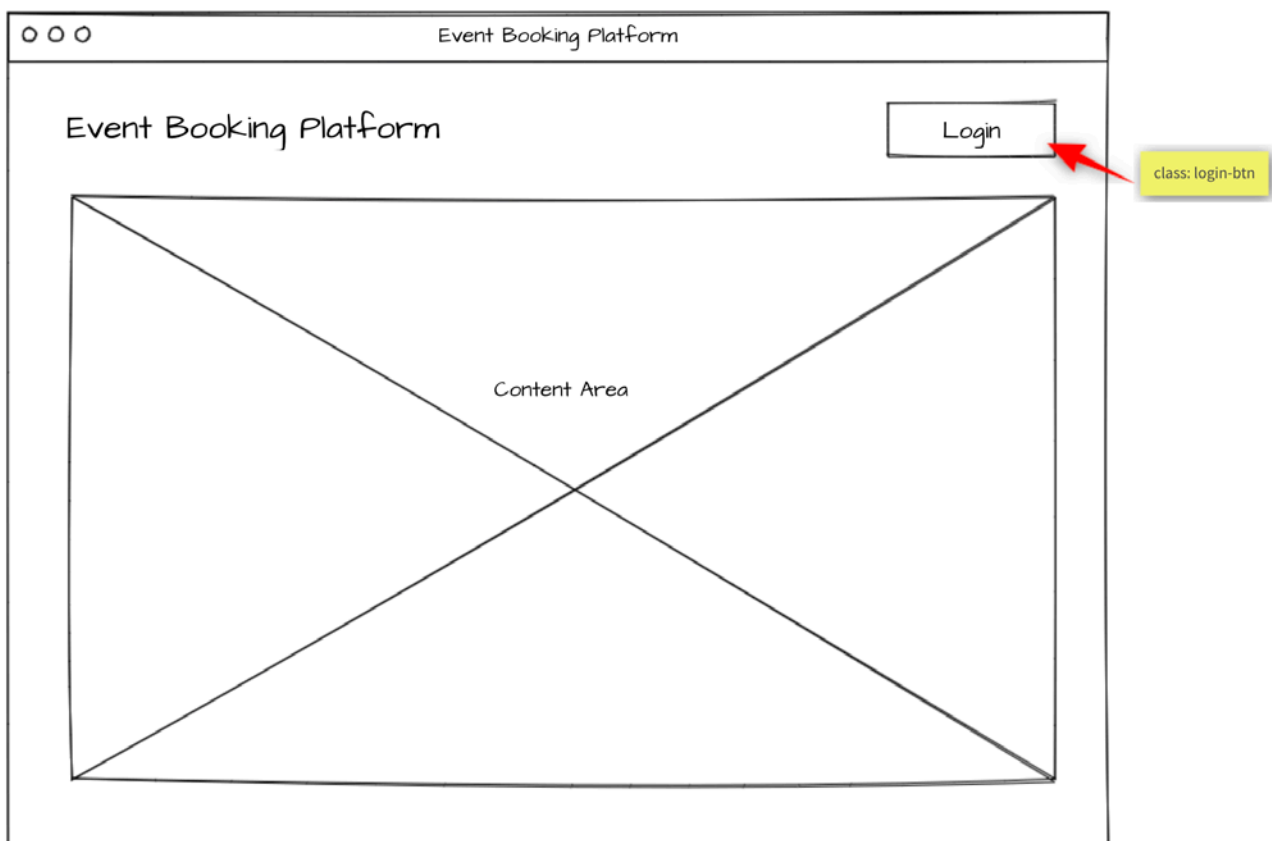
Interaction should show error/feedback messages based on response from the API where appropriate. Also, an error page should be shown when accessing an invalid URL.

Browsing all data is public and no user is required. But as soon as they want to register for an event, they need to be signed in.

Refreshing a page should always display the same page and it must be possible to share a page by URL so that all state is kept in the URL.

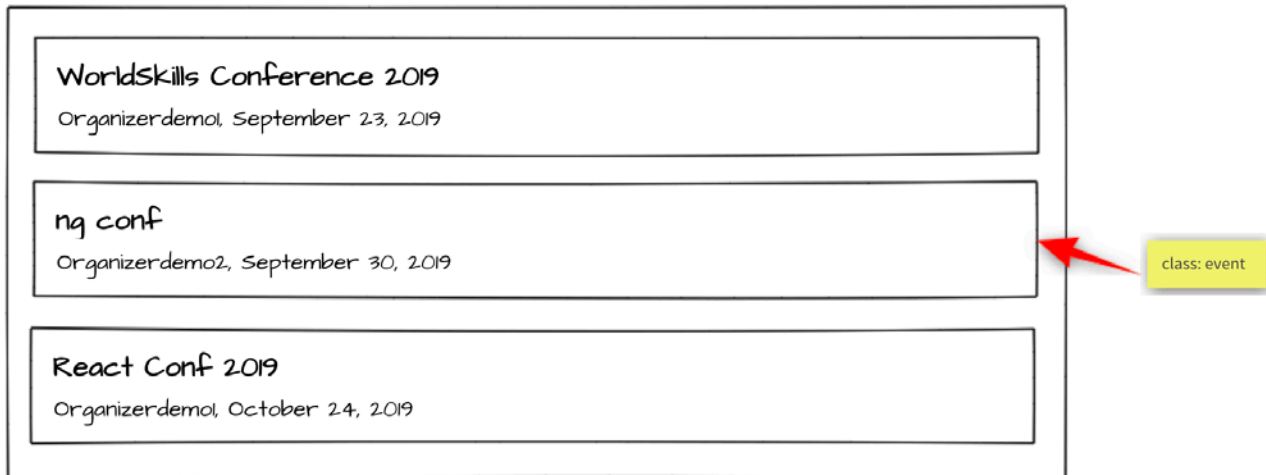
The layout should follow mockups exactly. The position of elements in design should be same as mockups.

The yellow bubbles in the mockups show classes and ids which have to be used for those elements to allow running automated tests. The yellow bubbles are not part of the design and do not have to be displayed in the browser.



Events list

When accessing the index page, all upcoming events are displayed with their organizer name and the date it takes place. By selecting an event, the event agenda will be shown.



Specification:

Feature C1 - List upcoming events: As an attendee, I want to see what events I can attend.

Scenario: C1a - List upcoming events

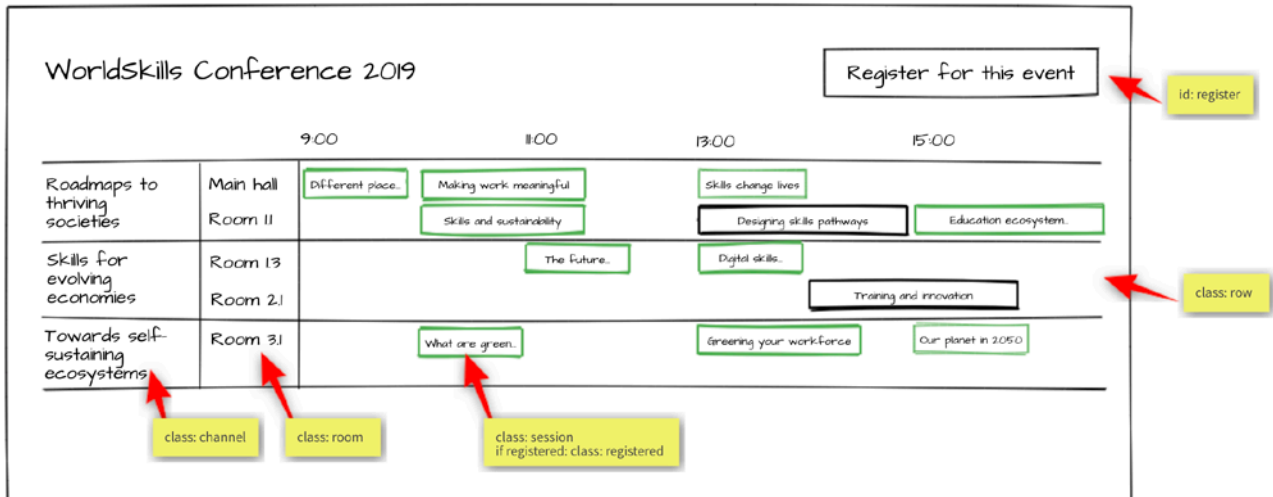
Given I am on the home page

Then I see a chronologically ordered list of upcoming events across all organizers

And I can click on an event

Event agenda

The agenda of the selected event is shown. If the user is logged in and is already registered for this event, all sessions with type talk are highlighted with a green border since they are automatically included in every event registration. Additionally, if the user signed up for addition sessions of type workshop, they will also be highlighted with a green border.



Specification:

Feature C2 - Display event agenda: As an attendee, I want to see what sessions are within an event.

Scenario: C2a - Display channels

Given I am on an event detail page

Then I see the channels of the event forming horizontal swimlanes

And the channels contain the rooms as "inside"-swimlanes

Scenario: C2b - Display talk and workshop sessions

Given I am on an event detail page

Then I see the timeline containing "9:00", "11:00", "13:00", "15:00"

And I see the sessions in the correct room swimlane starting and ending at the correct time

Scenario: C2c - Display registered sessions

Given I am on an event detail page

And I am logged in

And I am already registered for this event

Then I see all sessions of type talk highlighted with a green border

And I see all sessions of type workshop which I have signed up for highlighted with a green border

Session detail

When clicking on a session within the agenda, a detail page showing all information about this session will be shown.

Education ecosystem for the future - Workshop

In our fast-changing world, traditional forms of education are becoming outdated. We need to redesign our education systems. What are approaches to learning work in this new world?

Speaker: Pavel Luksha

Start: 15:00

End: 17:00

Cost: 30.-

Specification:

Feature C3 - Display session details: As an attendee, I want to inform myself about the details of a session.

Scenario: C3a - Display session details

Given I am on an event detail page

When I click on a session

Then I see a detail page of the session containing title, description, speaker, start and end time, type and the cost (if it is set)

Event registration

After clicking on "register for this event", the user either sees the event registration form if he is logged in or gets redirected to the login form if he is not logged in.

On the event registration form, he can select the ticket type he wants. Unavailable tickets are greyed out and can't be selected. The API is responsible for this information and the validation, so it does not have to be handled in the frontend.

The user can also select additional sessions of type workshop he wants to attend. If a session has an additional cost, it will show up in the cost view and add to the total cost.

After clicking on "Purchase", the user will be registered for this event and the sessions he has selected and gets redirected back to the event detail page.

WorldSkills Conference 2019

<input checked="" type="checkbox"/> Normal Ticket 210.-	<input type="checkbox"/> Early Bird unavailable 110.-	<input type="checkbox"/> Hotel Package 300.- 50 tickets available
---	---	--

Select additional workshops you want to book:

- Designing skills path
- Education ecosyste
- Training and innovat

Event ticket:	210.-
Additional workshops:	30.-
Total:	240.-

Specification:

Feature: C4 - Register for event: As an attendee, I want to be able to register for an event and pay the admission fee.

Scenario: C4a - Enter registration mode

Given I am on an event detail page
And I am logged in
Then I can click on a button to enter registration mode
And I see the ticket types and their cost
And I see a checkbox list of sessions of type workshop
And I see a cost view summary
And I see a disabled "Purchase" button

Scenario: C4b - Select ticket success

Given I am in registration mode of an event
When I select a ticket that is available
Then the ticket cost is displayed in the cost view summary
And the "Purchase" button is enabled

Scenario: C4c - Select ticket fail

Given I am in registration mode of an event
When I select a ticket that is not available
Then the ticket is not selected
And the "Purchase" button is not enabled

Scenario: C4d - Select workshop session

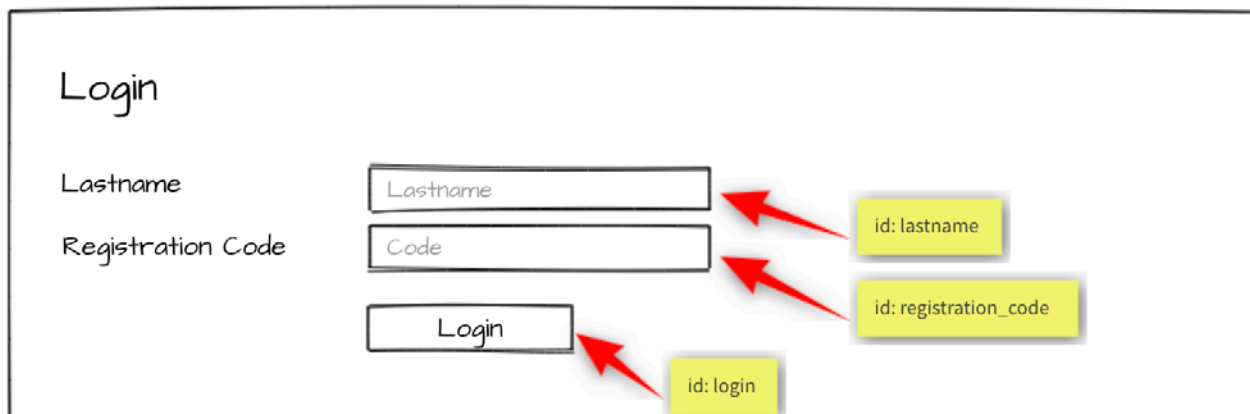
Given I am in registration mode in an event
When I tick a workshop session
Then the session receives a tick
And the price of the workshop is displayed under "Additional workshops" in
in the cost view
And the price of the session is added to the total cost

Scenario: C4e - Purchase

Given I am in registration mode in an event
When I select a ticket and I press the "Purchase" button
Then the registration is saved
And I get redirected to the event agenda
And a message appears "Registration successful"

Attendee login

When registering for an event, a login is required. After a successful login, the login button in the navigation should no longer be visible. Instead, the username of the currently logged in user should be displayed and a button to log out.



Specification:

Feature: C5 - Log in and log out as an attendee: As an attendee, I would like to log in and log out to the event platform so that I can access my registrations and register for new events.

Scenario: C5a - Attendee login ok

Given I am on the "login" page

When I enter the specified lastname and registration code for an attendee

And press the "Login" button

Then I'm forwarded to the previous page

And username and "Logout" button become visible in the header

Scenario: C5b - Attendee login fail

Given I am on the "login" page

When I enter a wrong lastname and registration code for an attendee

And press the "Login" button

Then I stay on the "login" page

And I see an error message "Lastname or registration code not correct"

Scenario: C5c - Attendee logout

Given I am logged in

When I press the "Logout" button

Then I am forwarded to the "login" page

And username and logout button disappear

And I cannot reach the previous page by going back

Notes

- When you are done with the first phase resp. after lunch you may continue with the second phase.
- If you are done with part one before the lunch break, you can start with part two by using your own API and exchange it at the start of the afternoon with the provided API.
- Competitors must use one of the server-side and client-side frameworks/libraries that are provided.
- The provided template for the admin interface should be used. It can be enhanced to get better functionality for your site. The visual design of the *admin interface* will *not* be assessed.
- The design of the attendee interface can be based on the template of the admin interface. The visual design of the attendee interface *will* be assessed: a simple, clean and attractive design is expected, including choice of colors, typography and spacing.
The design should be optimized for a minium window width of 1024px.
- For the admin interface it is important to use the selectors (class, id) as provided in the template. For the attendee interface it is important to use the selectors (class, id) as specified in the screen mockup (yellow bubbles).
- The specified database tables need to be implemented as provided. More tables may be added if needed. The given tables must not be changed. Provide a final SQL-dump and ERD screen as specified below.
- The API implementation should fulfill all requirements as stated in the description. All prefixes, RESTful-URLs and HTTP-Methods from the given API links must be implemented correctly and cannot be changed.
- Changes made in the data need to be propagated between frontend and backend in both directions.
- Update the first two organizers in the provided dump so it is possible to login to the system during assessment:
 - Organizer accounts:
 - Email: demo1@worldskills.org Password: demopass1
 - Email: demo2@worldskills.org Password: demopass2
- For the room capacity bar chart, you can use one of the provided JS-libraries (Charts.js, D3.js)

Submission Instructions to the Competitor

Morning: Files to be collected after the first phase on the server:

- Web service ([http://<hostname>/XX_JS-PHP_A/api/v1/..](http://<hostname>/XX_JS-PHP_A/api/v1/))
- Admin area (http://<hostname>/XX_JS-PHP_A/) - initially shows login page
- Folder inside webroot (XX_JS-PHP_A/)
- ERD screenshot named "XX_ERD.png" in "db-dump" folder inside of XX_JS-PHP_A
- Database dump named "XX_database.sql" in "db-dump" folder inside of XX_JS-PHP_A

Afternoon: Files to be collected after the second phase on the server:

- Front-end website (http://<hostname>/XX_JS-PHP_B/)
- Folder inside webroot (XX_JS-PHP_B/) - put all development files into XX_JS-PHP_B_dev.zip and leave only files for production-use in this directory.
- If the front-end root (http://<hostname>/XX_JS-PHP_B/) does not open the main page directly, have it open an index.html with a link to the main page for marking your front-end.

XX is your country code

The Web Service / REST-API will be assessed with the **automated test-framework** provided. Additional data may be used to test dynamic behavior.

It is recommended that you run the automated tests for the REST-API after implementing it locally or at least after uploading it to the server. Instructions to setup and use the tests for the API: see README.md in tests_api-php.

Manual assessment of other parts will be done using **Google Chrome**.

Equipment, machinery, installations, and materials required

It is expected that all Test Projects can be accomplished by Competitors based on the equipment and materials specified in the Infrastructure List*.

If any equipment or material is required that is not listed in the Infrastructure List, then it has to be listed here (in the table below - column headings may vary from skill to skill.)

WorldSkills International needs to approve any additional infrastructure requirements that are not listed in the Infrastructure List. Column headings may vary from skill to skill.

* (Definition: the Infrastructure List is the equipment, machinery, installations and materials supplied by the host country – it does not include tools and material to be supplied by Competitors and/or Experts)

ITEM	DESCRIPTION
Media Files	Contains: - DB dump - Template files for the admin-gui - Charts.js & D3.js incl. Documentation - Files for testing REST-API

Marking Scheme

SECTION	CRITERION	JUDG. MARKS	MEAS. MARKS	TOTAL
C1	Database	4.75 <u>3.85</u>	4.53	3.25 <u>6.85</u>
C2	Admin Backend	0	3.25 <u>6.5</u>	3.25 <u>6.5</u>
C3	Ticket and Session	0	4.53	4.53
C4	Organizer	42.35	2.25 <u>4.5</u>	3.25 <u>6.85</u>
C5	REST API	4.53 <u>3.35</u>	2.25 <u>4.5</u>	3.75 <u>7.85</u>
D1	Events list and agenda	0	3.75 <u>7.5</u>	3.75 <u>7.5</u>
D2	Session and Events	0	36	36
D3	General	42.35	36	48.35
D4	JS Quality	2.55 <u>3.35</u>	0	2.55 <u>3.35</u>
D5	Front end design	36.35	0	36.35
TOTAL		40.75 <u>23.6</u>	20.541	31.25 <u>64.6</u>

Framework

JS

- VueJs 2 (<https://github.com/vuejs-templates/webpack>)
- Angular 8
- React 16.12 (npx create-react-app my-app)

PHP

- Laravel 6

Software Workstation

- Windows 10
- PHPStorm
- WebStorm
- Sublime (No extension)
- Atom (No extension)
- VSCode (No extension)
- Node 12.13
- PHP 7.3 + phpunit
- Photoshop + Dreamweaver 2019
- MySQL Workbench
- Filezilla Client
- Chrome
- Postman

Software Server

- ESXi
- CentOS 7
- PHP 7.3 + phpunit
- Apache 2
- Node 12.13
- MariaDB 10.4
- PhpMyAdmin 4.9
- SSH
- FTP Server