

A10: Product

GROUP1714, 29/04/2018

- Daniel Ribeiro de Pinho - up201505302@fe.up.pt
- Francisco José Sousa Silva - up201502860@fe.up.pt
- Rui André Rebolo Fernandes Leixo - up201504818@fe.up.pt
- Vitor Emanuel Fernandes Magalhães - up201503447@fe.up.pt

1. Installation

The release can be found [here](#).

In order to test the Docker Hub image, one can run the following command:

```
docker run -it -p 8000:80 -e DB_DATABASE=lbaw1714 -e DB_USERNAME=lbaw1714  
-e DB_PASSWORD=kl12eu91 lantaros/lbaw1714:latest
```

2. Usage

The product can be found over at <http://lbaw1714.lbaw-prod.fe.up.pt>.

2.1. Administration Credentials

The website administrators can access the administration menu using the hyperlink on the nav bar or on <http://lbaw1714.lbaw-prod.fe.up.pt/administration>.

Username	Password
admin	pass

2.2. User Credentials

Type	Username	Password
Member	user	pass
Member	user2	pass

3. Application Help

Users receive help mainly in three ways:

- The most visible one is the FAQ page, accessible on the page footer, as well as there being links to it on the main page.
- Secondly, there is some online help included in some forms; this help is marked with a small (?) next to the field's name.

- Finally, there's a third way of getting help, through the Contact page. Users can fill out a form and the administrators will receive an email with the User's message.

In order to minimize the need for these help methods, we tried to make the website as intuitive as possible, and we feel like these ways go a great length in assisting the user.

4. Input Validation

On server-side, most of the input data is validated using Laravel's Validator, by checking restrictions as being null or not, the size of the string, etc. On client-side, the input data is validated by using Regex(pattern attribute) and the required attribute, as well as other HTML attributes.

For example, when creating an event:

- On client-side, the attributes required and maxlength are used in the following inputs, as well as the type of the input itself(like type="date" or "type=time"

```
<input id="name" name="name" required type="text" style="width:100%;"
      class="form-control col-xs-3 is-invalid" placeholder="Enter
event name" maxlength="80">

<textarea id="description" name="description" required class="form-
control"
          placeholder="Description" maxlength="255"
          style="width:100%;"> </textarea>

<input id="startDate" name="startDate" required type="date"
       style="margin-right:0.25em" class="form-control col-lg-8"
       style="width:100%;">

<input id="startTime" name="startTime" required type="time"
       class="form-control col-lg-3" style="width:100%;">
```

- On server-side, the function `edit` on `EventController` validates every data, by calling the `validator` function, with some validation rules.

```
protected function validator(array $data) {
    $validate = Validator::make($data, [
        'name' => 'required|string|max:64',
        'description' => 'required|string|max:516',
        'startDate' => 'date_format:"Y-m-d"',
        'startTime' => '',
        'endDate' => 'date_format:"Y-m-d|after_or_equal:startDate',
        'endTime' => '',
        'country' => 'required|string|exists:country,name',
        'ispublic' => 'boolean',
        'address' => 'nullable|string|max:50',
        'lodgingLink' => 'nullable',
    ]);
}
```

```
return $validate;  
}
```

5. Check Accessibility and Usability

The results of the accessibility test can be found in the file [acessibilidade.pdf](#), while the usability test results are in [usabilidade.pdf](#).

6. HTML & CSS Validation

The results of the HTML validation test (run on the Event page) can be found in the [event-html.pdf](#) file, while the CSS tests can be found in the [event-css.pdf](#) file.

7. Revisions to the Project

Our project evolved a lot since the beginning of the semester; we realized that some features would have to be either cut down or scrapped altogether; some of these features include flagging comments and posts, as well as interacting with events in more ways other than writing comments (such as using polls and uploading files).

However, we feel like the original idea still remained, and we did not deviate too much from the original idea of having a website based on finding the best events around.

8. Implementation Details

8.1. Libraries Used

We used several libraries and frameworks when developing this website:

In order to build the website structure, we used Docker, Laravel and Bootstrap; the effects of these frameworks can be found throughout the entire website.

Additionally, we used the FontAwesome, Material Icons and Google Fonts libraries in order to display more fonts than the system ones, as well as simple icons to accompany text; these can be seen in page titles, headers and buttons.

Finally, in order to show the event location maps on their pages, we used the Google Maps API to embed a small map with a pin to the event's address.

8.2 User Stories

8.2.1. User

Identifier	Name	Priority	Team members	State
U01	Basic Search	High	Everyone	100%
U02	FAQ Page	Medium	Everyone	100%
U03	Contact Page	Medium	Everyone	100%

Identifier	Name	Priority	Team members	State
U04	About Page	Low	Everyone	100%

8.2.2. Visitor

Identifier	Name	Priority	Team members	State
V01	Login	High	Everyone	100%
V02	Register	High	Everyone	100%
V03	Home Page	High	Everyone	100%
V04	Facebook Login	Low	n/a	0%

8.2.3. Member

Identifier	Name	Priority	Team members	State
M01	Home Page	High	Everyone	100%
M02	View Event	High	Everyone	100%
M03	Create Event	High	Rui Leixo, Francisco Silva	100%
M04	Create a Community	High	Daniel Pinho, Vitor Magalhães	100%
M05	Invite Member to a public Event	High	Francisco Silva	100%
M06	Logout	High	Vitor Magalhães	100%
M07	Advanced Search	Medium	Everyone	100%
M08	Access Private Events	Medium	Everyone	75%
M09	Comment Events	Medium	Everyone	100%
M10	Purchase Tickets	Medium	Rui Leixo	100%
M11	Advertisements	Medium	n/a	0%
M12	Add a friend	Low	Daniel Pinho	100%

8.2.4. Event Administrator

Identifier	Name	Priority	Team members	State
EA01	Edit Event	High	Everyone	100%
EA02	Delete Event	High	Rui Leixo, Francisco Silva	100%
EA03	Flag a comment	Medium	n/a	0%
EA04	Kick a Member	Medium	n/a	0%

8.2.5. Community Administrator

Identifier	Name	Priority	Team members	State
CA01	Invite a Member to a Community	High	Vitor Magalhães, Daniel Pinho	100%
CA02	Kick Member	High	n/a	0%
CA03	Delete Community	Medium	n/a	0%

8.2.6. Admin

Identifier	Name	Priority	Team members	State
A01	Flag Event	Medium	n/a	0%
A02	Flag Comments	Medium	n/a	0%
A03	Flag a Member	Medium	n/a	0%
A04	Home Page	Medium	Everyone	100%