

Locastic

We help clients create
amazing web and mobile apps

Assignment

Your assignment will be to build a simple web-shop which sells workshops.

All of the needed details of the requirements can be found in the User Stories document and the provided [design](#).

Depending on the role you are applying for you can solve the assignment in different ways:

- If you are applying as a React developer, you can solve the functional requirements without sticking strictly to the provided design, but it is important that all of the functionality is implemented correctly
- If you are applying for the UI/CSS developer role you can solve the assignment by implementing the design perfectly, but hardcoding all of the content and using simple JS for the needed interactions, without using a JS framework and connecting to the provided API at all
- If you are proficient and comfortable with both React and UI/CSS you can solve both parts fully which would give you an advantage over other candidates in whatever FE position you are interested in
- If the whole assignment is not finished by the deadline, feel free to send us the part that is done, and if the the quality of the work is sufficient, you will still be considered for the interview

Requirements

You can find the descriptions of our requirements for different roles below :

REACT/JS dev:

JUNIOR:

- Possible to use CRA (create-react-app) or custom setup
- Use ES6 syntax
- Adding types with Typescript
- Adding state management
- Adding a spinner when information is loading
- Deal with errors coming from the backend
- Divide the application in different pages and use a router
- Deploy the project somewhere

MID & SENIOR:

- Sustainable project architecture (folders structure, state management handling etc.)
- Code quality and performance
- Show common tools used for your daily development environment (linters, code formatter, git workflow and so on)
- Some sort of Unit Test
- Some sort of Integration Test

CSS/UI dev

JUNIOR:

- in-depth understanding of CSS and semantic HTML
- js basics (sliders, galleries, off-canvas navigation, etc.)
- ability to implement a design which behaves properly on every device and every browser
- experience with CSS preprocessing (sass, less, postCSS)
- basic working knowledge of build processes
- familiarity with different CSS methodologies (smacss, ITcss, OOCSS, BEM)
- experience in working with CSS frameworks
- experience with working with some form of dynamic web sites (js frameworks, CMS, or similar)
- familiarity with browser dev tools

MID & SENIOR:

- ability to plan and structure large projects keeping performance, maintainability, and scalability in mind
- knowing how to choose, set up and maintain different methodologies which are suitable for the project at hand

- communicate and work with designers in order to set up and maintain design systems
- strong understanding of performance
- strong knowledge of build processes, code and asset optimisations
- in-depth experience with browser dev tools (identify performance bottlenecks, debug a complex issue, inspect remote devices, working with local storage, cookies, etc.)
- intermediate js knowledge, js performance, more complex js based ui functionality, js based animations, basic js frameworks understanding, etc.
- Ability to create accessibility optimised UI's
- wide experience in working within a variety of different CMS's, templating languages and frameworks
- Ability to guide and mentor junior developers

When evaluating your UI/CSS solutions we will give scores based on the following points:

- Accuracy of cross browser / cross device design implementation
- Code structure and organisation
- Performance
- Accessibility

API Docs

There are two types of entities:

- User
- Workshop

User is the owner of the Workshop. Workshop is connected with the User using `userId` property which is equal to User's `id` property. For complete API documentation please refer to the link:

<https://documenter.getpostman.com/view/3243953/Szzn6Gmn?version=latest>

API Setup

1. Installation

NPM

```
npm install -D json-server
```

Yarn

```
yarn add -D json-server
```

2. Prepare the files:

Download and place the [db.json](#) file in your working directory

3. Start the server

Start this command from the working directory

```
json-server --watch db.json
```

This command will start the server at the localhost:3000 and will start the watcher so you don't have to manually refresh the server every time you make the change.

In case you need something additional from the json-server library you can find the docs here - <https://github.com/typicode/json-server>.

User Stories

General criteria valid for all user stories:

- Pixel perfect design
- Full responsiveness
- All functionalities defined in the documentation should be implemented
- Tested on different screen sizes and browsers (Chrome, Firefox, Safari, IE)

Specific story-related criteria will be mentioned next to the appropriate story.

Story	<p>In order to have an easily accessible Cart and link to the Workshop list, Customer should always see a navigation bar at the top of the page that contains two elements:</p> <ol style="list-style-type: none">1. Logo that works as a link to homepage2. Cart Icon that opens the Cart bar
Criteria	<ul style="list-style-type: none">● Sticky navigation - always at the top of the page when user scrolls● Logo links to Workshop list● Cart icon has a counter next to it that changes depending on the number of workshops added to the cart

Story	<p>In order to easily access his Cart, Customer should be able to trigger a Cart sidebar (from the navigation) with the following elements:</p> <ol style="list-style-type: none">1. Workshop item/s:<ol style="list-style-type: none">a. Workshop imageb. Titlec. Ticket Count (dropdown field)d. Pricee. Delete/Remove from cart Icon2. Total Price3. Checkout button
Criteria	<ul style="list-style-type: none">● Cart sidebar should act differently depending on screen size (check design for states)● Cart Icon and counter should move at the top of the sidebar as a title

Story	<p>In order to choose a Workshop, Customer should see a list of all available Workshops with the following elements:</p> <ul style="list-style-type: none">● Image● Category icon● Date● Time● Title● Price and currency● Button - add to basket
Criteria	<ul style="list-style-type: none">● By default workshops are listed in a chronological order from the closest to the farthest date from now● Inside the Workshop box, only the Image and Workshop Title should act as links to see more details about the Workshop● Button adds the item to the Cart (opens it) and changes the Cart Counter to notify the user● By default 9 workshops are listed - clicking on the Load more button should list the next 9 if available or all of the remaining if not

Story	In order to search through Workshops more efficiently, Customer should be able to filter Workshops by Category.
Criteria	<ul style="list-style-type: none">• Workshops from all categories are listed by default• Above the workshop list should be the title “Workshops” and beneath it the number of Workshops displayed for the selected category.• Each category has a specific icon that can be seen in the filter section and in each workshop box

Story	<p>In order to learn more details about a Workshop, Customer should see a Workshop details page with the following data:</p> <ul style="list-style-type: none">• Workshop image• Category Icon• Date• Time• Title• Speaker• Workshop description• Price per ticket• Number of tickets• Checkout (CTA button)• Total price for the selected amount of tickets
Criteria	<ul style="list-style-type: none">• Bonus: when the user clicks on the “<- Back” button it should ideally take him to the row in the workshop list where he clicked to see more details.• Adding a workshop to cart works the same way as on the Workshop list page. The only difference is that you can define the number of tickets straight away

Story	In order to see similar workshops to the one selected, Customer should have up to 3 recommended workshops at the bottom of the Workshop details page.
Criteria	<ul style="list-style-type: none">• Similar workshops are those which belong in the same category• If there is no other workshops from the same category, the section should not be displayed

Story	<p>In order to checkout desired workshop tickets, Customer needs to fill in a checkout form with the following fields:</p> <ul style="list-style-type: none">● Form Title and Description● First Name● Last Name● Email Address● Date of birth - datepicker● Gender - Select● Address● ZIP Code● "I Agree" checkbox● Checkout button
Criteria	<ul style="list-style-type: none">● Take into account placeholders and field validations

Story	<p>In order to know the Checkout process was successful, Customer should see a simple thank you page/modal with the following info:</p> <ul style="list-style-type: none">• Thank you - title• Description• Back to shop button
Criteria	<ul style="list-style-type: none">• Back to Shop button leads the user back to the default Workshop list page

Story	In order to know the owner of the shop, Customer should see Copyright info in the footer.
Criteria	