

Test task (NodeJS + React + Docker)

Time to complete - 8 hours

Write a client-server application in the blog format with 2 pages: list of articles + article. The application must run in Docker and consist of containers: client + server + database. Place the result on github

1. Customer

Libraries: React, React Router, Axios

Visual representation is arbitrary

Page: List of articles

1. Open by route '/' + '/articles'
2. Show article list
3. Sort the list by article ID in reverse order (the place of sorting implementation is arbitrary)
4. Put a title and creation date in the list item
5. By clicking on the title go to the page of the article in reading mode
6. Show New Article Button
7. By clicking on the button create an article and go to its page in edit mode

Page: Article

Reading mode:

1. Open on route '/article/{id}'
2. Show article title
3. Show creation date
4. Show edit date if article has been edited
5. Show article content
6. Show article delete button
7. By clicking on the button, delete the article and go to the page with the list
8. Show article edit button
9. Click on the button to switch to edit mode
10. Show button to return to list

Edit mode:

1. Open by route '/article/{id}?edit=true'
2. Show input with article title
3. Show textarea with article content

4. Show button to apply changes
5. By clicking on the button to save the article and switch to reading mode

2. Server

Libraries: Express.js, Knex.js

Routes:

/articles - GET return a list of articles
/article/:id - GET return the article
/article/:id - POST create article
/article/:id - PUT update article
/article/:id - DELETE delete article

Cover unit with tests

3. Database

SQL: PostgreSQL

Table article:

id - increments primary
heading - string length 255 default 'Untitled article'
content - string length 10000
created_at - timestamp nullable
updated_at - timestamp nullable