My API Blog

By Mathias Fortuna

Main Blog

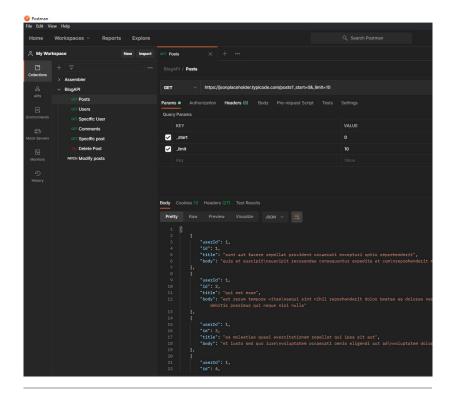
Main Features:

- Focused in functionality
- Responsive
- Infinite Scroll
- Lightweight
- Use of GET, DELETE and PATCH methods



HTML Organization

```
<body id="scrollableElement">
 <header class="navbar navbar-dark bg-dark shadow-sm sticky-top">...
 </header>
 <main>
   <div class="my-3">...
   </div>
   <section class="my-3" id="posts-container">...
   </section>
 </main>
 <!-- Posts Modal -->
 <div -
 </div>
 <!-- Edit Modal -->
 <div ---
 </div>
 <!-- Delete modal dialog -->
 <div --
 </div>
```



Postman usage

For testing prior code implementation

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
  Resources
  http://localhost:3000/posts
  http://localhost:3000/comments
  http://localhost:3000/users
  http://localhost:3000
GET /posts/5 200 32.140 ms - 225
GET /users/1 304 25.008 ms - -
GET /comments?postId=5 200 25.967 ms - -
GET /posts/4 200 23.370 ms - 270
GET /users/1 304 26.819 ms - -
GET /comments?postId=4 200 26.175 ms - -
GET /posts? page=1& limit=5 304 30.099 ms - -
GET /posts?_page=2&_limit=5 200 9.745 ms - -
GET /posts?_page=3&_limit=5 200 18.561 ms - -
GET /posts?_page=4&_limit=5 304 23.180 ms - -
GET /posts?_page=1&_limit=5 200 31.387 ms - -
GET /posts? page=2& limit=5 304 25.581 ms - -
GET /posts?_page=3&_limit=5 304 20.125 ms - -
GET /posts?_page=4&_limit=5 304 30.165 ms - -
GET /posts/3 200 29.949 ms - 283
GET /users/1 304 24.701 ms - -
GET /comments?postId=3 304 29.317 ms - -
DELETE /posts/3 200 44.462 ms - 2
GET /posts?_page=1&_limit=10 200 19.786 ms - -
GET /posts?_page=2&_limit=5 200 22.844 ms - -
GET /posts?_page=3&_limit=5 200 19.500 ms - -
GET /posts?_page=1&_limit=5 200 30.052 ms - -
GET /posts?_page=2&_limit=5 304 15.075 ms - -
GET /posts?_page=3&_limit=5 304 20.013 ms - -
GET /posts?_page=4&_limit=5 200 33.527 ms - -
GET /posts?_page=1&_limit=5 200 24.586 ms - -
```

Json Server

For successfully deleting or modifying data

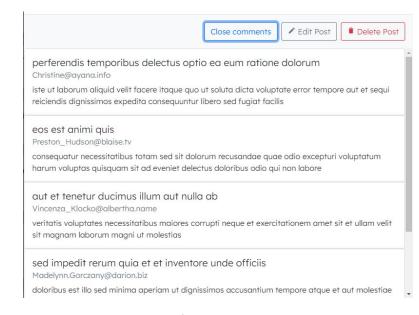


Cards





Modals and Buttons

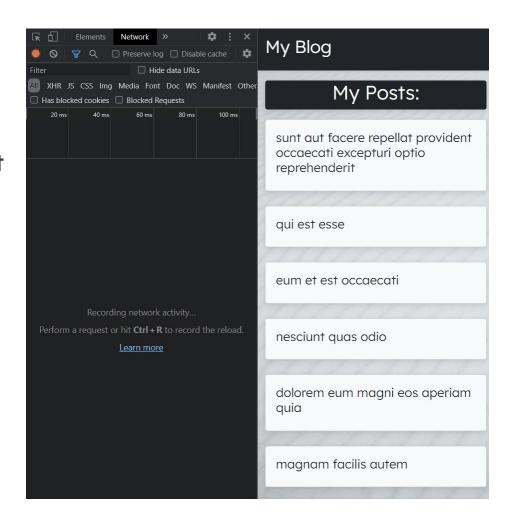


Collapsible element C Spinner

Bootstrap Elements

API Management

- Minimum memory usage by not saving requests
- Optimization of requests content upon needed information



Slow internet behaviour





Lessons learnt

- API first approach
- Bootstrap
- Postman
- Json Server
- Experience with documentation

Problems faced

- Data acquisition before elements injection
- Bootstrap unfamiliarity
- Live server behaviour

