

En la siguiente prueba técnica esperamos poder conocer como el desarrollador aborda ó toma desde una maqueta hecha en html y la procesa para generar un WebApp hecha en React. Nos gustaría conocer su ***pensamiento crítico y sistemático***, como también conocimiento y aplicación de su entendimiento de ***Paradigmas*** de programación.

Assets

Maqueta: <http://especiales.lanacion.com.ar/arc-css/acumulado.asp>

Estilos: <https://especiales.lanacion.com.ar/arc-css/css/site.css>

Endpoint:

<https://jrt2bb3b2nlkw5ozvfcl62wbe0pnifh.lambda-url.us-east-1.on.aws/> [get method available]

Objetivos

- A partir del tag main de html, componentizar todos los elementos que considere necesario.
- Debe respetar los estilos propuestos, nos concentramos más en observar cómo define y estructura los componentes
- El endpoint provisto contiene un json con lo que representan 30 notas o artículos. con estos datos se deben alcanzar los siguientes objetivos:
 - o Agrupar, totalizar y ordenar de mayor a menor los tags encontrados en las notas. Los tags serán encontrados en cada artículo dentro del atributo "taxonomy.tags"
 - o Del ordenamiento anterior mostrar los primeros 10 tags debajo del título **Acumulado Grilla**. El resultado de este item debe ser similar al siguiente resultado para ser mostrado en la lista

```
[
  {
    "slug": "huevo-tid47236",
    "text": "Huevo",
    "count": 13
  },
  {
    "slug": "leche-tid47244",
    "text": "Leche",
    "count": 9
  },
  {
    "slug": "arroz-tid47136",
    "text": "Arroz",
    "count": 7
  },
  {
    "slug": "manteca-tid47257",
    "text": "Manteca",
    "count": 7
  },
  {
    "slug": "azucar-tid47141",
    "text": "Azúcar",
    "count": 7
  },
  {
    "slug": "harina-0000-tid48184",
    "text": "Harina 0000",
    "count": 5
  }
]
```

```

    },
    {
      "slug": "ajo-tid47126",
      "text": "Ajo",
      "count": 4
    },
    {
      "slug": "pescados-tid67216",
      "text": "Pescados",
      "count": 4
    },
    {
      "slug": "frutas-tid67217",
      "text": "Frutas",
      "count": 4
    },
    {
      "slug": "semillas-de-sesamo-tid47338",
      "text": "Semillas de sésamo",
      "count": 3
    }
  ]

```

- o El href a contener debe ser: `"/tema/[tag.slug]"`
- o Tomar de imagen para el artículo la que sale en el `"promo_items.basics.url"`
- o Mostrar los 30 artículos en la grilla de artículos
 - listar sólo artículos con el `"subtype": "7"`,
 - la fecha a mostrar debe ser la de `display_date`
 - el formato de la fecha debe salir como sale en la maqueta
 - el título a mostrar se encuentra en el atributo `"headline.basic"`

Se espera observar:

Una web app hecha en React from Scratch o bien usar Next.js

La grilla y los artículos deben venir renderizados desde el servidor.

Implementar Hooks de ser necesario

Repositorio con el código en GitHub

No usar librerías de terceros

Uso de programación declarativa

Test unitarios

Será un plus si agrega (aunque es opcional)

Para manejo de estado entre componentes, si lo considera necesario aplicar Context API de React .

Implementar una API Client patterns para consultas del Endpoint.

Usar docker o bien implementar el código resultante en alguno Paas gratuito como heroku o now.sh, etc.