

Anda ditugaskan untuk membuat Single Page Application (SPA) menggunakan ReactJS yang terdiri dari module:

1. Halaman utama berupa posts list yang bisa diakses oleh semua pengunjung.
2. Halaman admin yang didalamnya akan memiliki fitur data management berupa blog **posts** berupa JSON format yang akan diambil dari <https://jsonplaceholder.typicode.com/posts>. Referensi api endpoints bisa dicek di sini <https://jsonplaceholder.typicode.com/guide/>.

Sebelum bisa mengakses halaman admin dan melakukan Create, Update, dan Delete pada posts, tentunya, user harus berstatus logged in terlebih dahulu.

## Rules

1. **(PLUS)** Anda diperbolehkan menggunakan Salah satu diantara kedua UI library ini: ChakraUI atau Material UI sebagai ui component framework
2. Gunakan React Router untuk setiap perpindahan view/screen
3. Gunakan axios untuk setiap ajax requests
4. Gunakan Redux untuk state management
5. Components yang akan digunakan untuk tasks dibawah harus reusable dengan html templating seminimal/seefisien mungkin
6. Dilarang menambahkan lagi third party library lain
7. Wajib mengoptimalkan usability dari SPA dengan menambahkan keyboard events listener sesuai context yang sedang aktif berjalan (esc, enter, tab, arrows, dsbg..)
8. **(PLUS)** Improvement tampilan, penambahan transition animation, class naming

## Global Tasks

1. Buat halaman login (<http://localhost:3000/login>) dengan UI Popup dialog untuk mensimulasikan auth process menggunakan **userId** dan **email** sebagai login parameter, simpan data logged in user pada localStorage. (gunakan beberapa userId/email pada response di sini <https://jsonplaceholder.typicode.com/users/>) endpoint login akan menggunakan get request ke endpoint <https://jsonplaceholder.typicode.com/users/{userId}>
2. Buat permission check untuk halaman admin dan halaman login.
  - a. Halaman login akan otomatis redirect ke halaman admin jika user sudah berhasil login.
  - b. Halaman admin akan otomatis redirect ke halaman login jika user belum login.
  - c. Tombol logout mesti tersedia secara global dan dapat diakses di halaman manapun untuk user yang sudah berhasil login.
6. State management pada nomor 7 dan 8 di atas harus menggunakan Redux Jika Anda menggunakan UI library, Override CSS bawaan secara global:
  - a. Font family: Nunito ([https://fonts.google.com/specimen/Nunito?preview.text\\_type=custom](https://fonts.google.com/specimen/Nunito?preview.text_type=custom))
  - b. Grid breakpoint:
    - md: 760px
    - lg: 900px
  - c. Max width container: 960px untuk semua css media query (xs, sm, md, lg)
  - d. Primary color: #106cc8

## Tasks Pada halaman utama

1. Halaman berupa card lists dari post item yang diambil dari endpoint <https://jsonplaceholder.typicode.com/posts> (<https://jsonplaceholder.typicode.com/posts?page=1&limit=5>)
2. Pagination menggunakan konsep infinite scroll, halaman berikutnya akan otomatis bertambah jika scroll down sampai scroll height hampir habis
3. Visitor bisa melakukan like/dislike untuk liked post pada masing-masing post item, data liked post items akan disimpan di localStorage
4. Liked post items akan diakses di halaman <http://localhost:3000/liked-posts>
5. UI untuk read post item akan berupa popup dialog menampilkan comments dari post item tersebut. (<https://jsonplaceholder.typicode.com/comments?postId=1>)
6. **(PLUS)** Gunakan nested route untuk modal, jadi meskipun UI akan berupa modal popup, saat read post item tertentu dilakukan, route/url pada browser tetap akan berpindah secara normal sebagai berikut:  
**Read post id 1**, url menjadi <http://localhost:3000/posts/1>

## Tasks Pada halaman admin

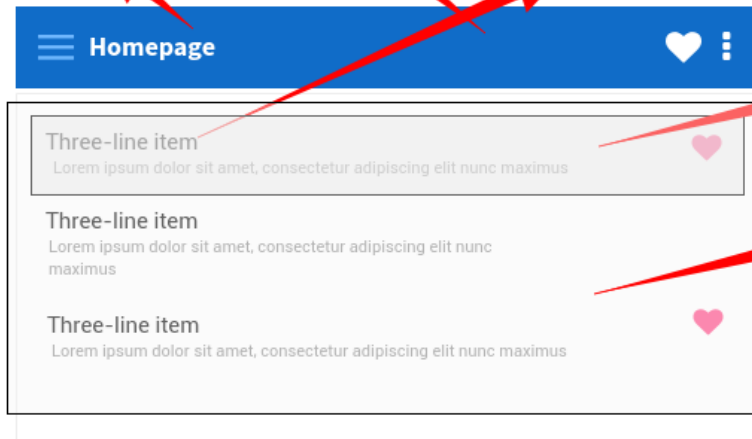
1. Pada url <http://localhost:3000/admin/> akan menampilkan posts. Masing-masing post item berelasi pada user tertentu, jadi anda diharuskan hanya menampilkan list posts yang dimiliki oleh user yg sedang login pada SPA tersebut.
2. Meskipun UI akan berupa dialog, saat create/edit dilakukan, route/url pada browser tetap akan berpindah secara normal sebagai berikut:  
**Editing post id 1**, url menjadi <http://localhost:3000/admin/posts/1/edit>  
**Creating new post**, url menjadi <http://localhost:3000/admin/create>  
**Read post id 1**, url menjadi <http://localhost:3000/admin/posts/1>
3. Delete post item, sebelumnya mesti menampilkan terlebih dahulu confirmation dialog.
4. Fitur create/edit harus menggunakan shared component yang sama berupa form dialog, field validation wajib ditambahkan
5. **(PLUS)** Fitur read post harus menggunakan shared component yang sama dengan yang digunakan di halaman utama

## Halaman utama

Font weight: 600  
color: #fff

Color tag: primary

Color: #000  
Font size: 18px



Gunakan css flex box: untuk centering items dan memisahkan antara bagian kiri (content) dan bagian kanan (toggle like button)

Cards menggunakan pseudo css untuk membuat striped color: ganjil background #fff, genap background #eee

**Login**

userId

1


Email\*



ipsum@lorem.com

Cancel

Login

## Halaman Admin

 Admin



Three-line item


Lorem ipsum dolor sit amet, consectetur adipiscing elit nunc maximus

Three-line item


Lorem ipsum dolor sit amet, consectetur adipiscing elit nunc maximus

Three-line item

Lorem ipsum dolor sit amet, consectetur adipiscing elit nunc maximus



Create Post



Title

\*\*\*\*\*

Body

\*\*\*\*\*

SAVE

## Lorem Ipsum

Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
Nunc maximus, nulla ut commodo sagittis, sapien dui mattis  
dui, non pulvinar lorem felis nec erat

Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
Nunc maximus, nulla ut commodo sagittis, sapien dui mattis  
dui, non pulvinar lorem felis nec erat

Lorem ipsum dolor sit amet, consectetur adipiscing elit.  
Nunc maximus, nulla ut commodo sagittis, sapien dui mattis  
dui, non pulvinar lorem felis nec erat

### Comments

**Two-line item**  
Secondary text

**Two-line item**  
Secondary text

**Two-line item**  
Secondary text

CLOSE