

## Bài tập ReactJS

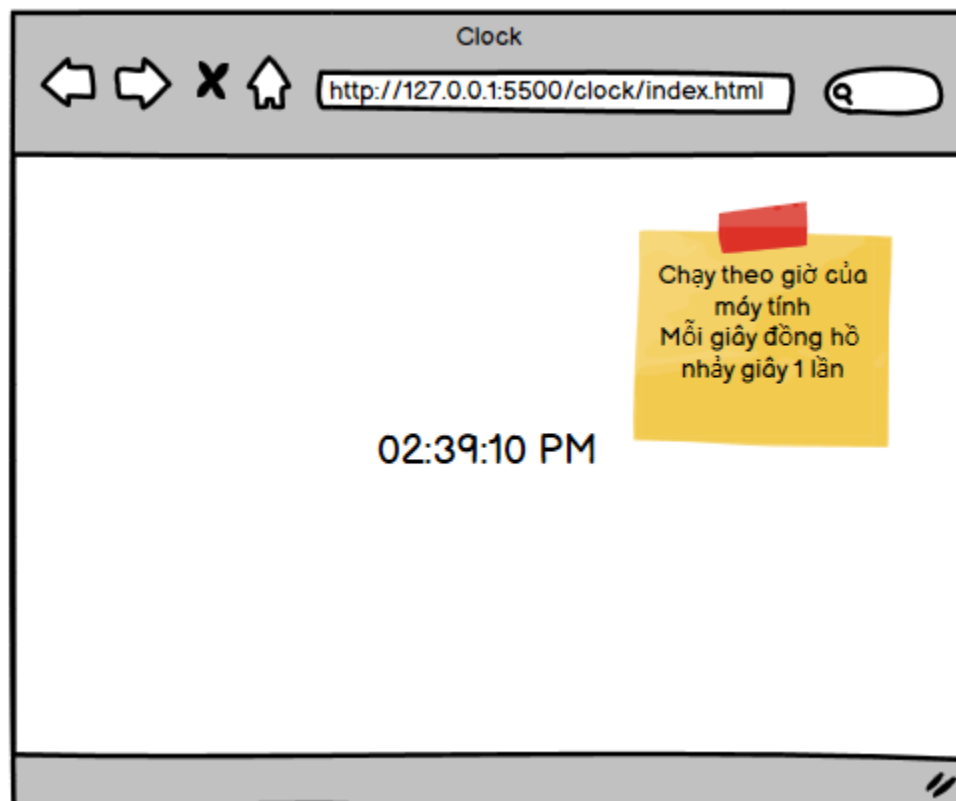
### 1. Xây dựng 1 clock như hình dưới, làm việc với Object Date

Mặc định session là AM

Nếu giờ là 0 thì giờ là 12

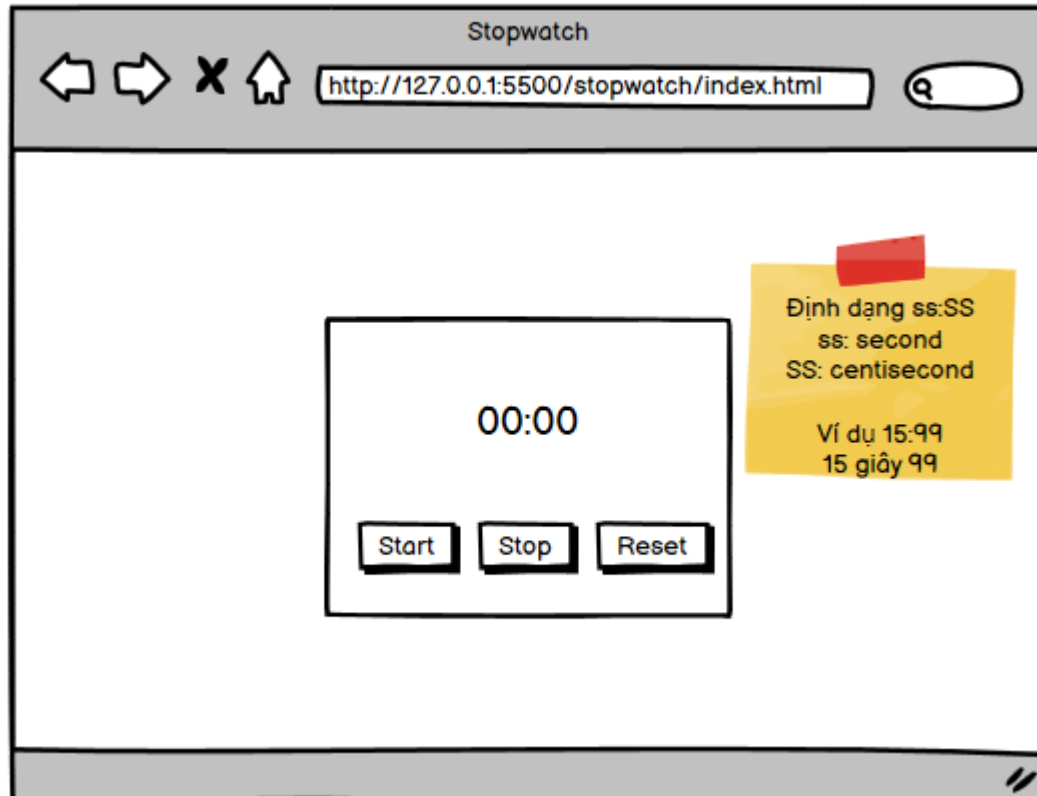
Nếu giờ lớn hơn 12 thì giảm đi 12, đồng thời session là PM

Sử dụng setTimeout, useState, useEffect



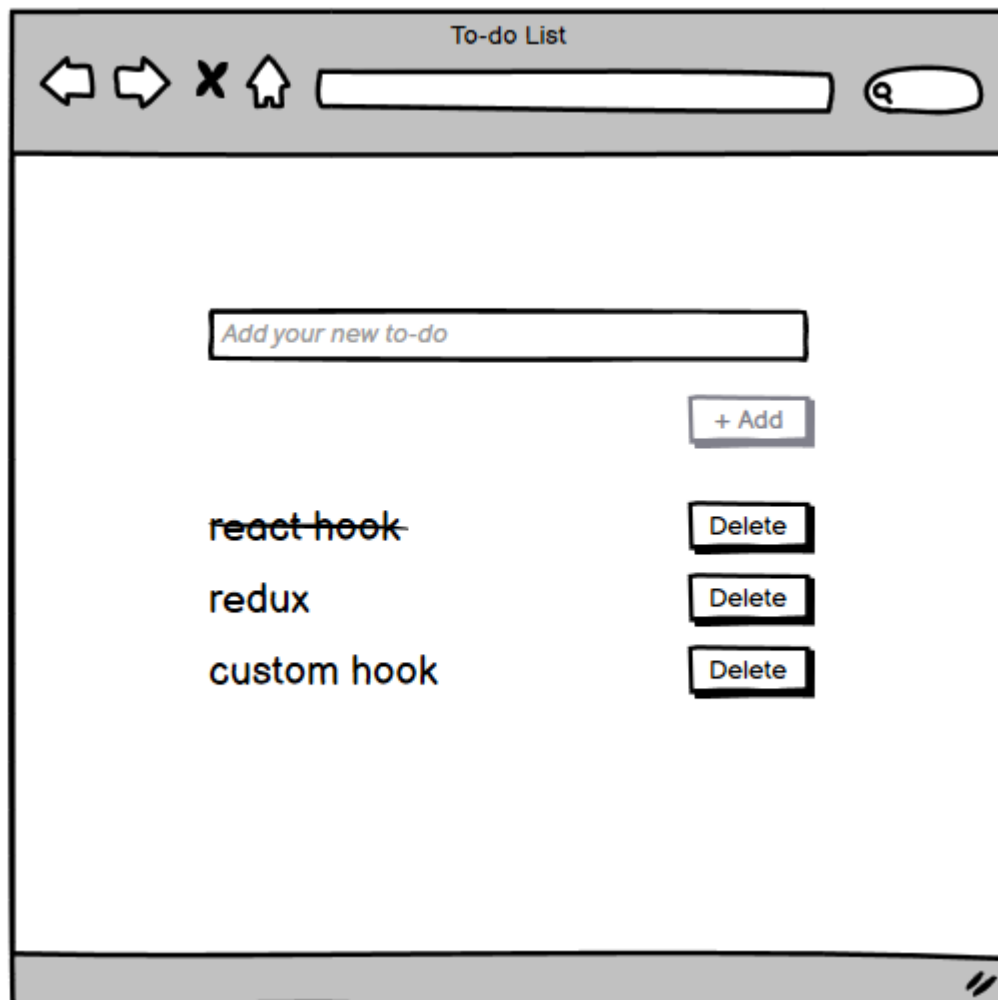
## 2. Xây dựng 1 stopwatch như hình dưới

Sử dụng `setTimeout`, `useState`, `useEffect`, `useRef`



### 3. TodoList

- `1. Xây dựng giao diện như hình dưới
- `2. Khi click nút Add, hoặc Enter input, nội dung input sẽ được lưu vào localStorage, sau đó input sẽ được clear
- `3. Khi chạy trang lần đầu, hoặc reload trang, hiển thị danh sách to-do
- `4. Khi click nút Delete, hiển thị dialog xác nhận có xóa hay không, nếu “yes” thì xóa
- `5. Nếu input rỗng, hiển thị toàn bộ danh sách, khi input thay đổi, tiến hành lọc danh sách
- `6. Nếu input rỗng hoặc input trùng với 1 to-do có sẵn, disable nút Add
- `7. Mỗi to-do có trạng thái đã hoàn thành hoặc chưa, nếu to-do có trạng thái đã hoàn thành, to-do sẽ bị gạch ngang như hình dưới. Ngược lại, có thể click vào to-do để hoàn thành



#### 4. Weather App

`1. Xây dựng giao diện như hình dưới.

`2. Khi click nút Forecast, sẽ fetch API và render giao diện dự báo thời tiết theo ngày như bên dưới. Mỗi item thể hiện thông tin thứ, ngày tháng (xem mục số 3), icon, nhiệt độ tối thiểu, nhiệt độ tối đa.

API url:

<http://api.weatherapi.com/v1/forecast.json?key=e0eecb964303463dbeb81526220411&q=Ho%20Chi%20Minh%20City&days=3&aqi=no&alerts=no>

Với param **q** là location input, param **days** là days input.

API Header:

```
{
  "Transfer-Encoding": "chunked",
  "Connection": "keep-alive",
  "Vary": "Accept-Encoding",
  "CDN-PullZone": "93447",
  "CDN-Uid": "8fa3a04a-75d9-4707-8056-b7b33c8ac7fe",
  "CDN-RequestCountryCode": "GB",
  "CDN-ProxyVer": "1.03",
  "CDN-RequestPullSuccess": "True",
  "CDN-RequestPullCode": "200",
  "CDN-CachedAt": "11/04/2022 08:16:51",
  "CDN-EdgeStorageId": "595",
  "CDN-Status": "200",
  "CDN-RequestId": "f307d41d8fbab99a4b97d485b7902431",
  "CDN-Cache": "MISS",
  "Cache-Control": "public, max-age=180",
  "Content-Type": "application/json",
  "Date": "Fri, 04 Nov 2022 08:16:51 GMT",
  "Server": "BunnyCDN-CZ-595"
}
```

`3. Sử dụng thư viện moment.js để định dạng ngày, giờ.

`4. Hiểu bản chất của useState và useRef, dùng useRef cho 2 input.

`5. Custom hook useFetch.

`6. Khi người dùng click vào một ngày bất kỳ, sẽ render ra giao diện dự báo thời tiết theo giờ như hình bên dưới. Mỗi item thể hiện thông tin giờ, icon, nhiệt độ, độ ẩm.

Weather

⬅️ ➡️ ✕ 🏠

🔍

Location

Ho Chi Minh City

Number of days

3

Forecast

Today 4

☔

29.5  
22.4

Sat 5

⚡

29.5  
22.4

Sun 6

☔

29.5  
22.4

Now

☔

T 25.5  
H 85

20 : 00

☔

T 25.5  
H 85

21 : 00

☔

T 25.5  
H 85

22 : 00

☔

T 25.5  
H 85

23 : 00

☔

T 25.5  
H 85

## 5. SnapShot App

Xây dựng một web app xem ảnh như bên dưới. Sử dụng nguồn ảnh từ ~~tumblr~~ flickr.

1. Người dùng submit input Search, lấy danh sách ảnh từ api sau

API URL:

```
`https://api.flickr.com/services/rest/?method=flickr.photos.search&api_key=${apiKey}&tags=${query}&per_page=24&format=json&nojsoncallback=1`
```

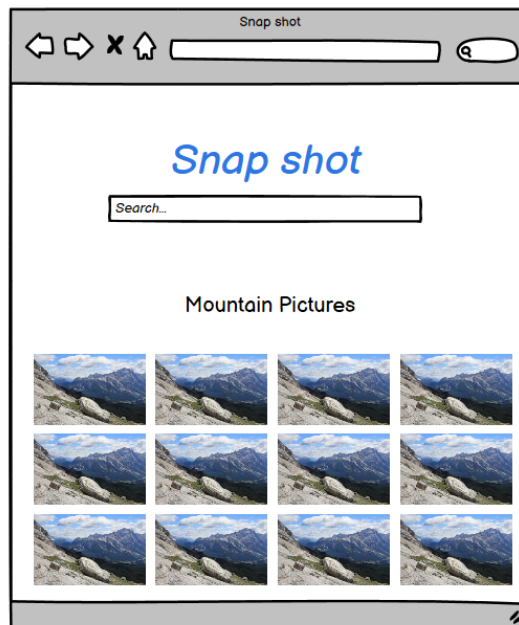
Trong đó, apiKey

```
export const apiKey = "636e1481b4f3c446d26b8eb6ebfe7127";
```

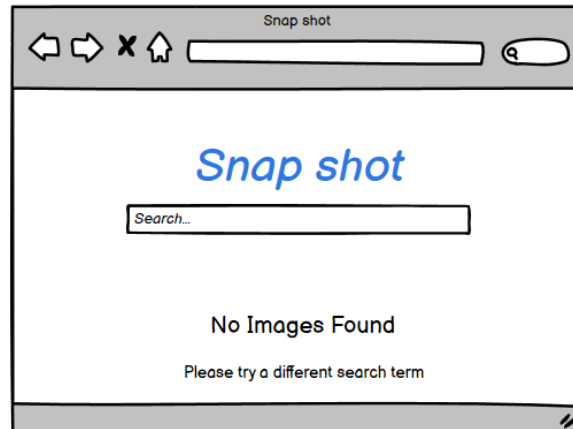
Image src:

```
`https://farm${farm}.staticflickr.com/${server}/${id}_${secret}_m.jpg`
```

Khi hover image, sẽ có hiệu ứng, ví dụ scale image 1.25



`2. Not found

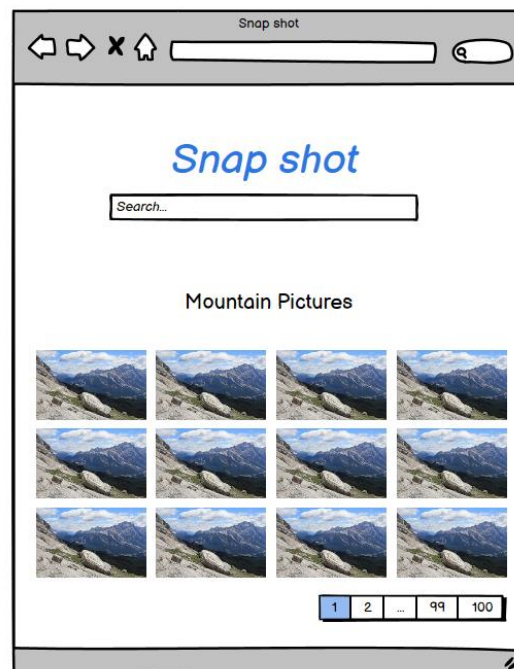


`3. Sử dụng hook useContext để xử lý các state globally

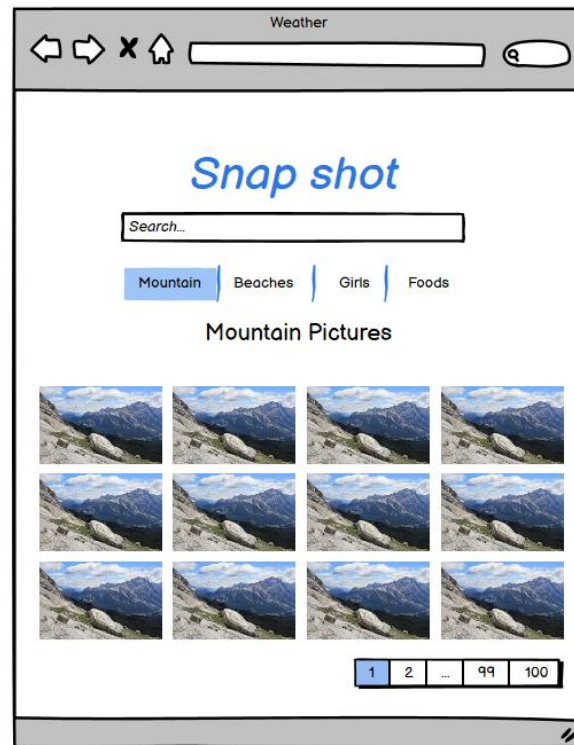
`4. Phân trang gallery

Thêm param **page** vào API URL

**Quy ước:** total page là attribute pages chạy lần đầu tiên



## `5. Thêm shortcut/navigation



## `6. Sử dụng thư viện react-router-dom

Khi người dùng search **soccer** khi đó url sẽ là **/search/soccer**

Khi chọn navigation, url sẽ là **/nav**, trong đó nav là mountains, beaches,...