Để áp dụng Redis caching cho REST API thì phải thỏa 2 tiêu chí

1. Tần suất truy vấn thường xuyên
2. Dữ liệu trả về của REST API ít thay đổi, nếu thay đổi liên tục (VD: product liên tục thay đổi thì ko được)
3. Dữ liệu trả về phải lớn (chứ trả về1, 2 dòng thì quá ít)

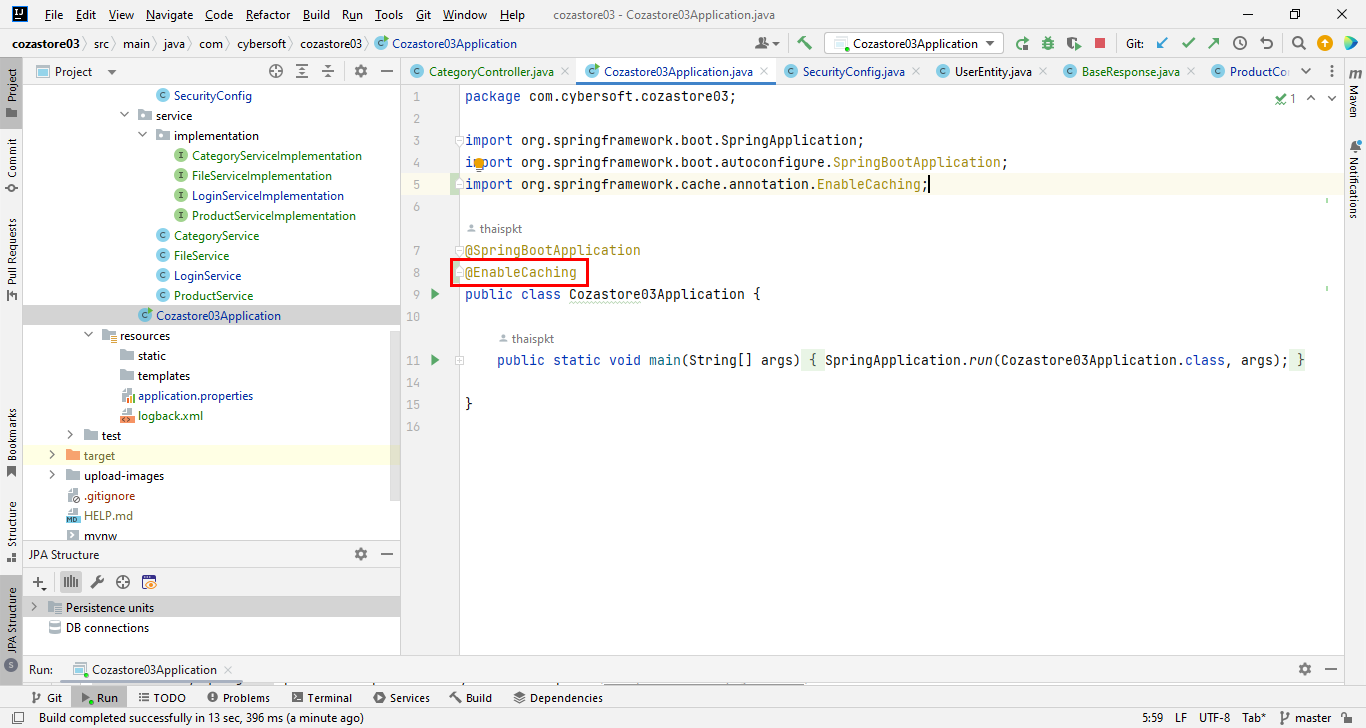
VD: Ở 1 số trang chính phủ, đầu tiên chọn tỉnh thành phố, sau đó chọn Quận trong list, trong mỗi quận lại chọn Phường, mỗi phường lại chọn đường -> khoảng 44K records

Solution: cache trên RAM -> truy xuất cực kì nhanh -> nhưng nếu server die -> mất data

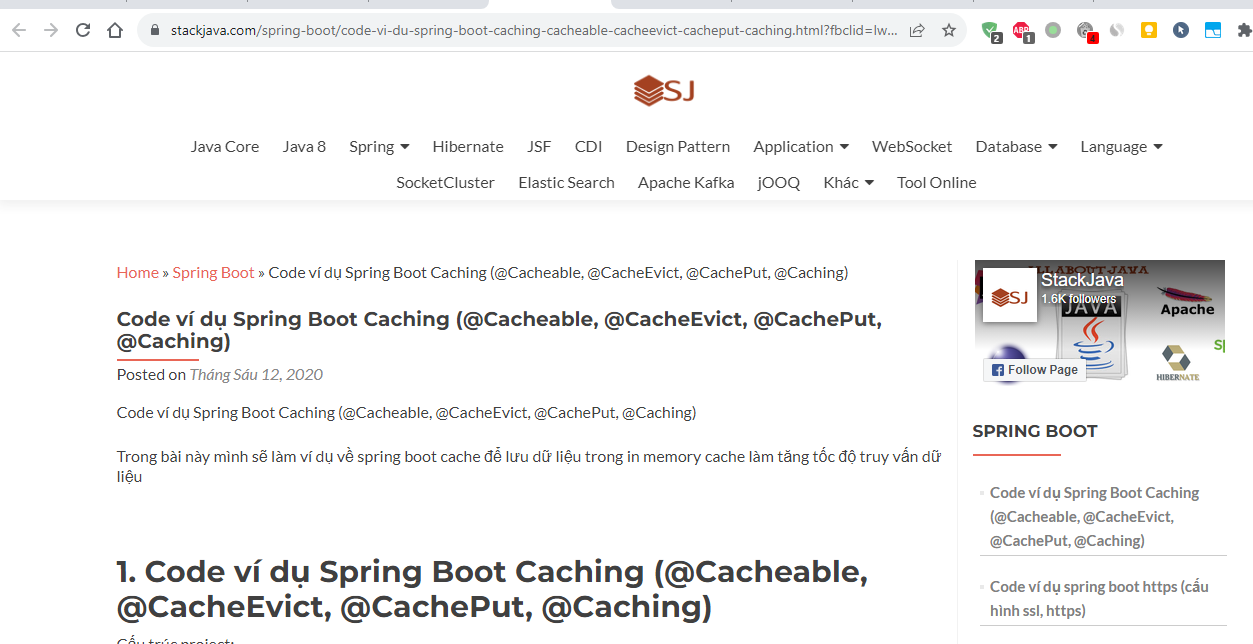
1. Dữ liệu ít thay đổi gọi là master data (đây là thuật ngữ bên dev) không đổi. Ít thay đổi cỡ 1-2 năm đổi 1 lần. VD: product-category table là bảng ít thay đổi.
2. Cache trên RAM, khi deploy multi-instance: 1 con instance chết thì 2 con còn lại ko dùng được mặc dù Redis có hỗ trợ share cache.

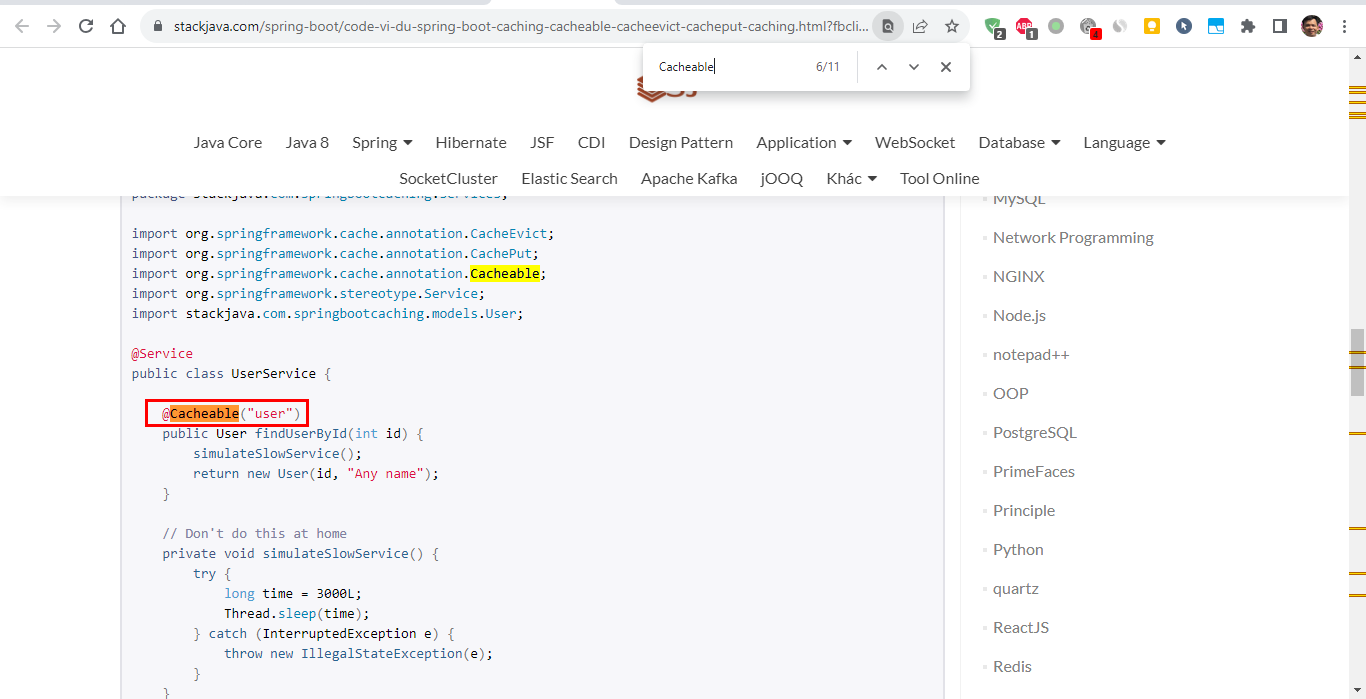
Khi clone app để làm load balancer

1. Tại sao Redis lại được ưu dùng ?
   1. Hỗ trợ lưu trữ key-value -> có thể truy xuất bằng JSON (ko phụ thuộc language, ko phụ thuộc của entity mapping
   2. Redis hỗ trợ lưu data trong RAM hoặc trong file. Tùy theo mức độ dữ liệu lớn thì lưu trên file, còn dữ liệu nhỏ thì lưu trên RAM.



<https://stackjava.com/spring-boot/code-vi-du-spring-boot-caching-cacheable-cacheevict-cacheput-caching.html?fbclid=IwAR2KbI32sjzxPGWZR6ON24ygxD4oBxNZgIqd2_Wk_ffUF_2wO0qFI7McAAI>





Nếu cache ở Repositor

Nếu như request ko truyền parameter -> cache ở Service

Nếu như request ko parameter động -> cache ở Repository

// Viet 1 API tra ra danh sach Category, roi cache no lai