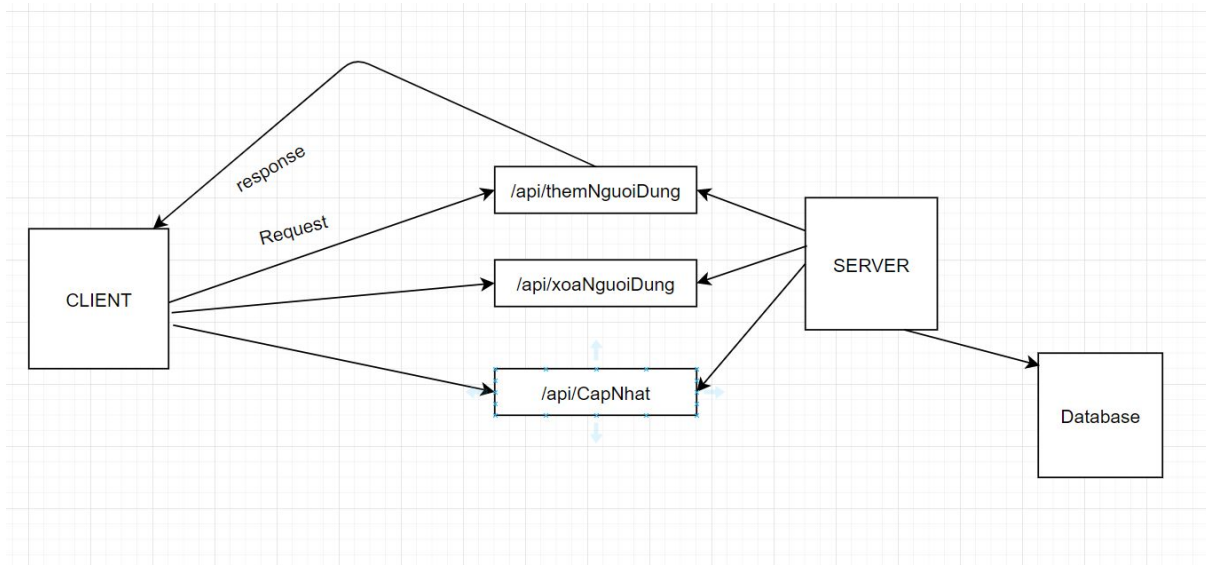


Tổng Hợp kiến thức Axios

1. Mô hình giao tiếp Front and Back



- ❖ Công việc của front end, chính là hiển thị data ra màn hình cho người dùng sử dụng. Vd: dữ liệu là danh sách 10 object phim => hiện ra màn hình item phim .
- ❖ Front end không thể lấy dữ liệu trực tiếp từ database
- ❖ Back end sẽ quản lý database, và viết cái đường dẫn, còn gọi là API, mỗi API tương ứng với 1 chức năng trên giao diện
 - Muốn lấy ds Người dùng, thì gửi request lên api “/api/layDanhSachNguoiDung” , back end sẽ xử lý vào trả về data là ds người dùng
 - Muốn lấy thêm một người dùng, vấn đề là ds người dùng nằm trong DB, từ client không thêm được, do đó phải gửi request lên api “/api/themNguoiDung” (hình trên), gửi kèm theo thông tin nhập từ form, back end sẽ xử lý vào trả về data là kết quả thêm được thay không

2.Http method: theo chuẩn Restful API

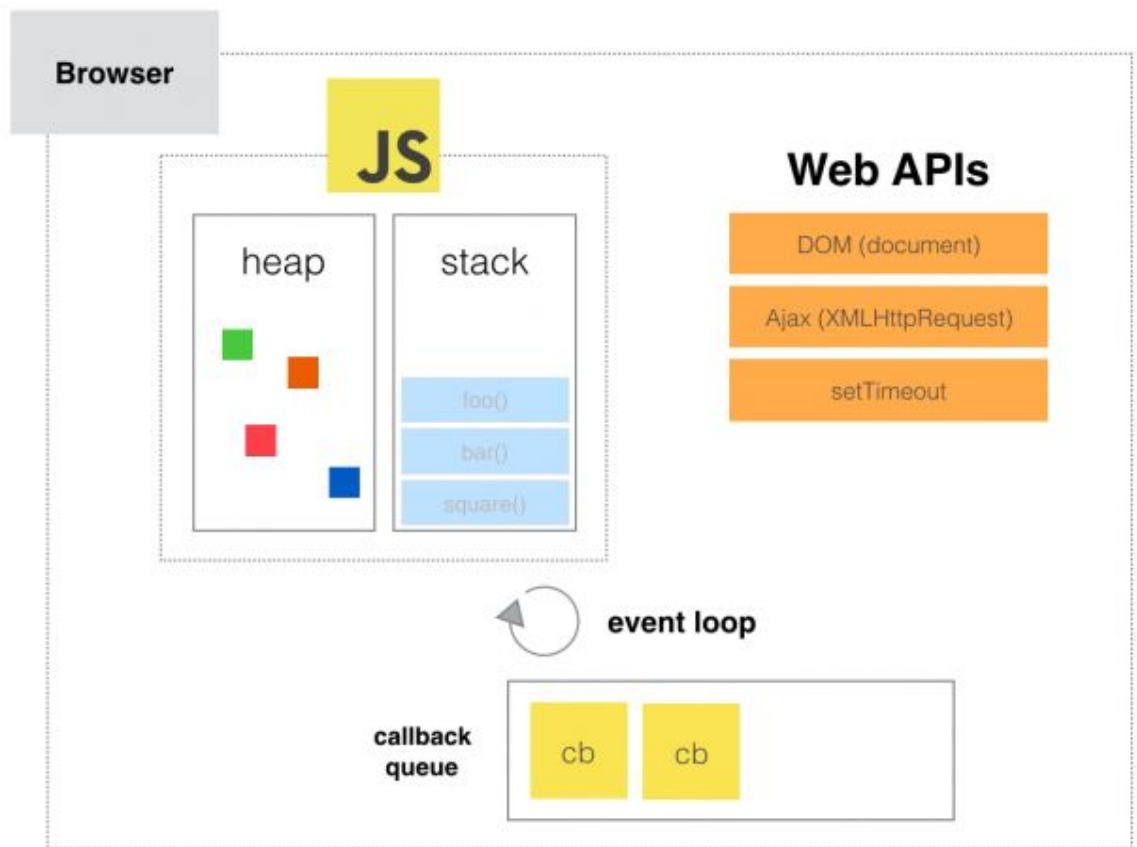
- ❖ GET: lấy dữ liệu
- ❖ POST: thêm dữ liệu
- ❖ PUT: cập nhật dữ liệu theo dạng ghi đè

❖ DELETE: xóa dữ liệu

❖ PATCH: cập nhật dữ liệu dạng thành phần, gửi lên thuộc tính nào thì chỉnh sửa thuộc tính đó

3. Axios : thư viện hỗ trợ gửi request xuống Backend, axios chạy trả về promise, có 2 phương thức : then(hàm xử lý) nếu thành công và catch(hàm xử lý) nếu thất bại

4. Asynchronous: mặc định code js được xử lý đồng bộ, tuy nhiên một số hàm để tranh dừng hệ thống khi tốn quá nhiều thời gian xử lý, sẽ được chuyển qua xử lý bất đồng bộ



Mô hình xử lý code của browser

- heap: lưu trữ variable trong call

- stack: nạp các câu lên vào stack chạy đồng bộ, xử lý xong sẽ rì mu khỏi stack
- web APIs: stack gặp lên asynchronous sẽ đẩy code qua đây để chạy riêng,
- callback queue: async nào chạy xong sẽ được đưa xuống hàng chờ
- event loop: quan sát Stack, khi nào stack clear thì đẩy async ở hàng chờ lên và chạy