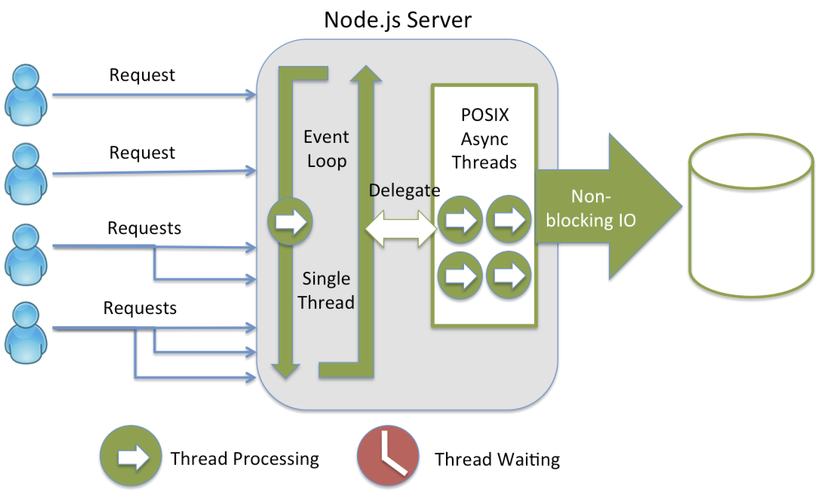
1. **NodeJS**

NodeJS là một platform cho phía server được xây dựng dựa trên Google Engine V8, với khả năng giúp lập trình viên implement phần backend của một hệ thống web application bằng Javascript.



1. **Event Emitter**

* Được hiểu như notification trong swift, khởi tạo 1 lắng nghe và 1 thông báo.
* Khi đối tượng EventEmitter phát ra một sự kiện, tất cả các hàm được gắn vào sự kiện cụ thể đó được gọi một cách đồng bộ.

var events = require('events');

var eventEmitter = new events.EventEmitter();

eventEmitter.on('clicked', function() {

console.log('Something is clicked!');

})

eventEmitter.addListener('clicked', function() {

console.log('Something is clicked!');

})

Ref: <https://viblo.asia/p/tim-hieu-event-va-eventemitter-trong-nodejs-bWrZnx2p5xw>

1. **NPM và NPMJS**

NPM là gì? NMP là viết tắt của Node package manager là một công cụ tạo và quản lý các thư viện lập trình Javascript cho [Node.js](https://nodejs.org/). Trong cộng đồng Javascript, các lập trình viên chia sẻ hàng trăm nghìn các thư viện với các đoạn code đã thực hiện sẵn một chức năng nào đó. Nó giúp cho các dự án mới tránh phải viết lại các thành phần cơ bản, các thư viện lập trình hay thậm chí cả các [framework](https://topdev.vn/blog/framework-la-gi/).

**NPM** là một phần mềm cài đặt trên máy tính của bạn nên bạn có thể sử dụng nó để cài đặt các thư viện **Javascript** từ trên **Internet**. Để cài đặt một thư viện nào đó, chỉ cần mở cửa sổ **Terminal (hoặc CMD)** và thực thi lệnh giống dưới đây:

npm install package-name

Ref: <https://www.npmjs.com>

1. **Cassandra**

Cài đặt cassandra qua Homebrew

Cài đặt Homebrew

ruby -e "$(curl -fsSL https://raw.github.com/Homebrew/homebrew/go/install)"

Cài đặt python

brew install python

Cài đặt sql

pip install cql

Cài đặt cassandra

brew install cassandra

Ref: <https://gist.github.com/hkhamm/a9a2b45dd749e5d3b3ae>

Tiến hành truy cập cassandra

Cài đặt cassandra-driver

npm install cassandra-driver

Ref: <https://www.npmjs.com/package/cassandra-driver>

Kết nối database

const cassandra = require('cassandra-driver');

const client = new cassandra.Client({

contactPoints:['127.0.0.1:9042'],

localDataCenter: 'datacenter1',

credentials: { username: 'tphuong', password: 'tphuong' }

});

client.execute('SELECT \* FROM tphuongdb.tbl\_sings', function(err, data) {

if (err) {

// handle error

console.log('handle error.', err)

} else {

console.log('handle success.', data)

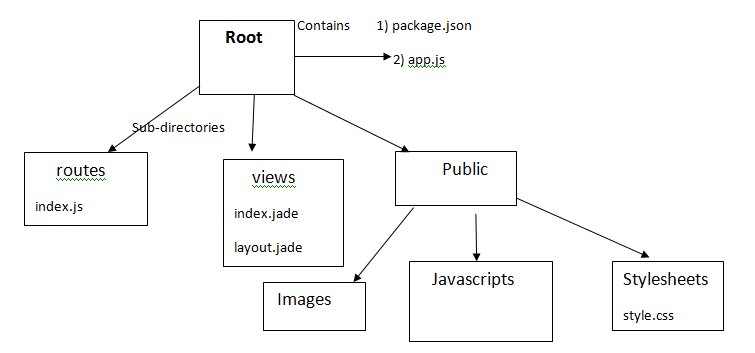
// handle success.

}

});

**5. ExpressJS**

**Expressjs** là một framework được xây dựng trên nền tảng của **Nodejs**. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. **Expressjs** hỗ trợ các method HTTP và midleware tạo ra API vô cùng mạnh mẽ và dễ sử dụng.



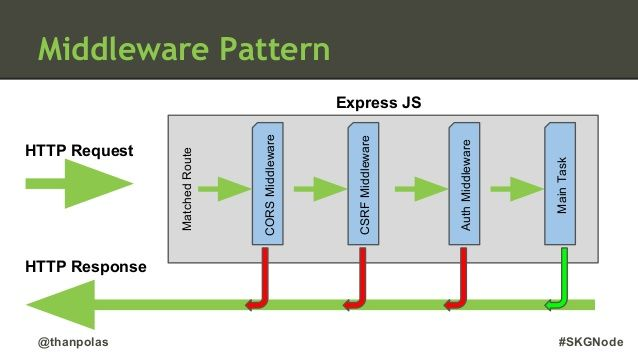
Install

npm install express --save

Ref: <https://topdev.vn/blog/express-js-la-gi/>

**6. Middleware**

**Middleware** là những đoạn mã trung gian nằm giữa các request và response. Nó nhận các request, thi hành các mệnh lệnh tương ứng trên request đó. Sau khi hoàn thành nó response (trả về) hoặc chuyển kết quả ủy thác cho một **Middleware** khác trong hàng đợi.



Hình trên mô tả 3 middleware có trong ExpressJS. Một request khi gửi đến Express sẽ được xử lý qua 5 bước như sau :

1. Tìm Route tương ứng với request
2. Dùng CORS Middleware để kiểm tra cross-origin Resource sharing của request
3. Dùng CRSF Middleware để xác thực CSRF của request, chống fake request
4. Dùng Auth Middleware để xác thực request có được truy cập hay không
5. Xử lý công việc được yêu cầu bởi request (Main Task)

Trong Express, có 5 kiểu middleware có thể sử dụng :

* Application-level middleware (middleware cấp ứng dụng)
* Router-level middleware (middlware cấp điều hướng - router)
* Error-handling middleware (middleware xử lý lỗi)
* Built-in middleware (middleware sẵn có)
* Third-party middleware (middleware của bên thứ ba)

Ref: <https://viblo.asia/p/tim-hieu-ve-middleware-trong-expressjs-gVQelwaaGZJ>