**Soal Nomor 1**

**Jelaskan Secara detail Ap aitu Node.js**

Node.js adalah sebuah perangkat lunak yang berfungsi untuk mengembangkan aplikasi pada web yang ditulis dengan menggunakan bahasa pemograman JavaScript. Seperti kita tahu bahwa, kebanyakan bahasa pemrograman JavaScript sering dipergunakan pada sisi *client* dalam hal ini adalah pada browser. Dengan adanya Node.js penggunaan bahasa pemrograman JavaScript bisa lebih luas lagi.

Jika menggunakan Node.js maka anda bisa menggunakan [bahasa pemrograman JavaScript](https://bootup.ai/blog/javascript-adalah-berikut-penjelasan-fungsi-dan-cara-kerjanya/" \t "_blank) pada sisi server seperti pearl, ruby dan PHB. Perlu diketahui bahwa Node.js bisa dijalankan di beberapa *operating system* seperti linux, Mac OS X dan tentunya pada windows. Di sini, adalah bahwa Node.js memiliki pustaka server khusus sehingga server web bisa berjalan layaknya NGINX dan [Apache](https://www.apache.org/).

**Perbedaan Javascript dan Node.js**

Node.js sudah banyak dipakai oleh perusahaan-perusahaan besar di dunia. Adakah perbedaan antara JavaScript dan Node.js? Secara umum, [bahasa pemograman](https://bootup.ai/blog/jenis-bahasa-pemrograman-dan-fungsinya/" \t "_blank) yang digunakan pada JavaScript dan Node.js sama. Hanya saja, Node.js memiliki kumpulan API yang berbeda antara API yang satu dengan API yang lainnya.

Dengan menggunakan Node.js, maka browser anda akan memiliki beberapa API DOM.WEB dimana semuanya terbuka sehingga memudahkan beriteraksi dengan UI dan mengakses hardware dengan batasan yang ada.

Karena Node.js dilengkapi dengan banyak API, maka hal ini sangat cocok untuk mengembangan banyak *back end* yang bisa didukung oleh permintaan https, proses anak, aus dan lain sebagainya. Sementara, browser memiliki berbagai dukungan dasar dimana masih bisa dibatasi demi alasan keamanan.

**Kelebihan Node.js**

Node.js memiliki banyak keunggulan sehingga saat ini sudah banyak digunakan secara luas. Keunggulannya di antara lain adalah sebagai berikut:

1. **Mudah dipelajari**

Pada tahun 2016 diadakan *user survey*. Dari hasil survey tersebut diketahui bahwa Node.js sangat populer digunakan oleh mereka yang mengembangkan *Front end*. Menurut para pengembang tersebut, mereka bisa dengan mudah memahami dan mengerti dengan bahasa universal. Oleh karena itu, mereka bisa dengan mudah menerapkan Node.js Pada *back end*. Dalam laporannya, disebutkan bahwa programmer javaScript Junior pun bisa belajar dan bekerja dengan cepat.

1. **Keberadaan Fullstack JS yang sangat memudahkan**

Sebelumnya, javaScript hanya digunakan pada sisi client. Dengan adanya Node.js fullstack, maka memudahkan para programmer untuk menulis pada *back and* dan *front end*.

1. **Komunitas yang sangat aktif**

Saat ini, banyak programmer yang tertarik dengan Node.js. Ada banyak komunitas-komunitas di luar sana yang akan memudahkan anda untuk mendapatkan solusi dari masalah-masalah yang mungkin timbul ketika anda sedang mengerjakan sebuah projek berbasis Node.js.

**Kekurangan Node.js**

Beberapa kekurangan dari Node.js di antaranya adalah sebagai berikut:

1. **API Node.js tidak stabil**

Salah satu kekurangan utama ialah adanya API yang tidak stabil. Hal ini menjadi perhatian utama para programmer. API Node.js seringkali berubah dimana perubahan tidak sesuai bisa memberi masalah tersendiri bagi para programmer. Seringkali programmer harus membuat beberapa perubahan vital pada basis kode supaya bisa *compatible* dengan sistem terbaru yang ada.

1. **Memerlukan waktu pengembangan yang lebih lama**

Banyak programmer yang mengeluh bahwa dengan menggunakan Node.js mereka memerlukan waktu lebih lama dalam mengembangkannya. Hal ini karena disebabkan banyak hal. Sebagai contoh, ketika anda menggunakan Ruby On Rain anda akan menemukan banyak petunjuk dalam hal melakukan sesuai.

Sedangkan pada Node.js programmer harus menulis sesuatunya dari awal. Hal ini yang membuat sesuatunya lebih lama. Dalam hal ini, banyak programmer yang mensiasatinya dengan bekerjasama bersama programmer lain untuk membuat dan mengembangkan kode secara lebih cepat dan efektif.

1. **Untuk Heavy-Computing Node.js tidak direkomendasikan**

Semua software pasti punya kekurangan dan kelebihan. Meski memiliki banyak kelebihan tetap saja ada hal-hal yang tidak cocok untuk project tertentu. Nah, Node.js ini ternyata tidak cocok untuk *heavy computing*. Hal ini disebabkan karena tidak bisa menghandle multi thread yang berat.

Meski bisa melayani aplikasi yang rumit, namun Node.js akan memerlukan waktu yang sangat lama. Pekerjaan berat seperti perhitungan berlebihan, memblokir permintaan yang masuk bisa secara signifikan menurunkan kinerjanya. Namun demikian, Node.js bisa bekerja dengan baik pada aplikasi kompleks, namun demikian kinerjanya akan melemah bila harus menghadapi komputasi berat.

1. **Tool yang bisa dibilang kurang matang**

Inti dari Node.js bisa dibilang stabil. Namun demikian, ada banyak paket registru NPM yang memiliki kualitas buruk. Paket NPM ini merupakan manajer paket yang mengatur pemasangan dan melakukan pengelolaan program dari pihak ketiga di luar Node.js.

Perlu diketahui bahwa sebagaian besar dari ecosystem Node.js adalah *open source*. Karena merupakan *ecosystem open source,* tentunya akan banyak tool yang tidak terawasi dengan baik. Dalam hal ini, maka kualitasnya tidak standard karena memang tidak ada yang melakukan pengawasan.

Contoh kasus lainnya, ada beberapa tools yang sebenarnya tidak lolos *standard* pengkodean tetapi tetap saja dipakai dan dipergunakan dalam *ecosystem*. Anda bisa meminta bantuan programmer yang berpengalaman untuk menemukan *tool* yang handal.

**Soal nomor 2**

**Aturan atau Cara Kerja Node.js**

Cara kerjanya berbeda dengan cara kerja bahasa pemrogramman server lainnya yang bersifat melakukan *blocking*. Sementara itu, Node.js memiliki cara kerja yang bersifat *Non-blocking*. Dalam hal bekerja, memiliki kesamaan dengan JavaScript, yaitu dalam hal menghandel *event* dimana mereka bekerja secara basis event baru selanjutnya dilakukan pengalihan ke kode pemrograman selanjutnya.

Sebuah Node.js bisa memiliki algoritma seperti di bawah ini:

* Halaman blog akan meminta request ke server
* Data diambil dari data base
* Data blog ditulis secara HTML
* Selanjutnya respon akan dikirim ke klien.

Bahasa pemrograman blocking memiliki sifat multi-thread berpengaruh besar urutan dalam hal pemrosesan dan permintaan data. Ketika program berjalan dan ketika pengambilan data dalam waktu tertentu maka threat berikutnya sudah harus disiapkan agar proses pengambilan data bisa berjalan dengan lancar.

Hal ini berbeda dengan yang dilakulkan dengan system yang menggunakan Node.js. Pada Node.js, threat hanya akan dibuat bila ada even yang membutuhkannya saja. Hal ini tentunya akan membuat proses pengolahan data bisa dilakukan dengan lebih efektif dan lebih cepat.

Software yang satu ini memungkinkan programmer melakukan pemasangan JavaScript bukan saja di bagian *client* browser tetapi juga pada bagian server. Ini adalah terobosan penting dalam dunia programming. Penggunaan Node.js semakin meluas. Banyak perusahaan-perusahaan terkemuka yang menggunakan Node.js untuk men-support aplikasi yang dikembangkannya. Node.js adalah solusi bagi beberapa perusahan untuk memberikan inovasi terbaru di dunia programming.