

BAB III

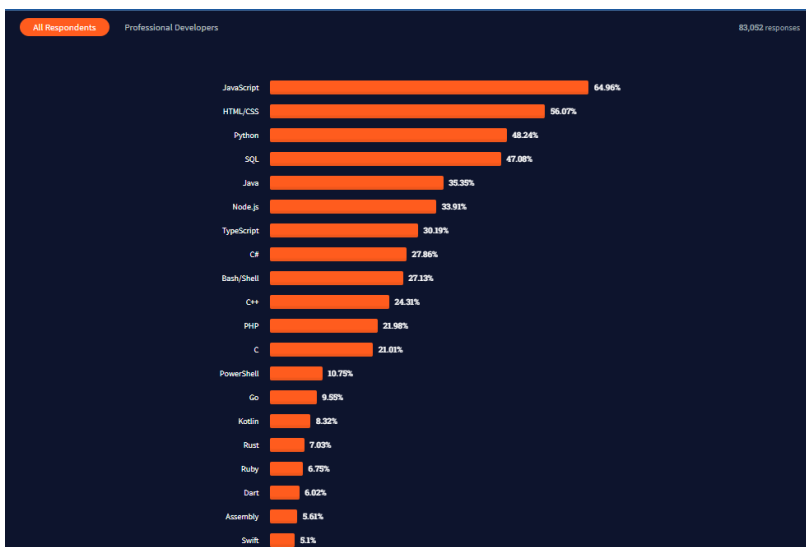
ANALISIS DAN PERANCANGAN

3.1 Analisis

3.1.1 Identifikasi Masalah

Menurut data survey dari (Insight Stack Overflow survey 2021, 2021) JavaScript menyelesaikan tahun kesembilan berturut-turut sebagai bahasa pemrograman yang paling umum digunakan. Bagi sebagian besar pengembang, pemrograman adalah pemrograman web. Sebanyak 64.96% dari 83,052 responden dari total seluruh pengembang yang mengikuti survey menggunakan bahasa pemrograman Javascript (Insight Stack Overflow survey 2021, 2021).

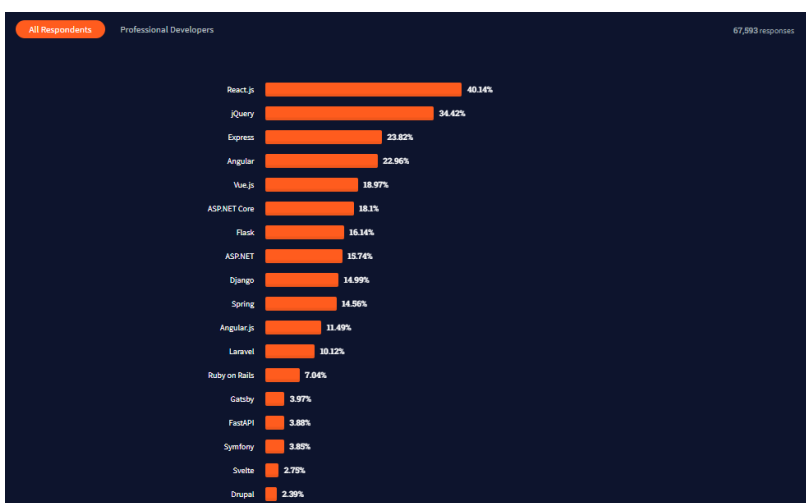
Commented [JP12]: Sebutkan sumbernya



Gambar 3. 1 Grafik trend bahasa pemrograman (Sumber: stackoverflow.com, 2021)

Dan dari *trend web framework tech* saat ini, 5 peringkat tertinggi mendukung bahasa pemrograman javascript, diantaranya: React.js, jQuery, Express, Angular,

dan Vue.js. Namun untuk jQuery penggunaannya mulai ditinggalkan karena method jquery yang biasa digunakan untuk manipulasi DOM mulai tergantikan dengan vanilla javascript yang lebih simple dan efektif (Gese, 2019). Di sisi lain, jQuery bukan merupakan jenis framework yang bisa dibandingkan langsung dengan React.js, Angular, dan Vue.js karena berbeda fungsi.

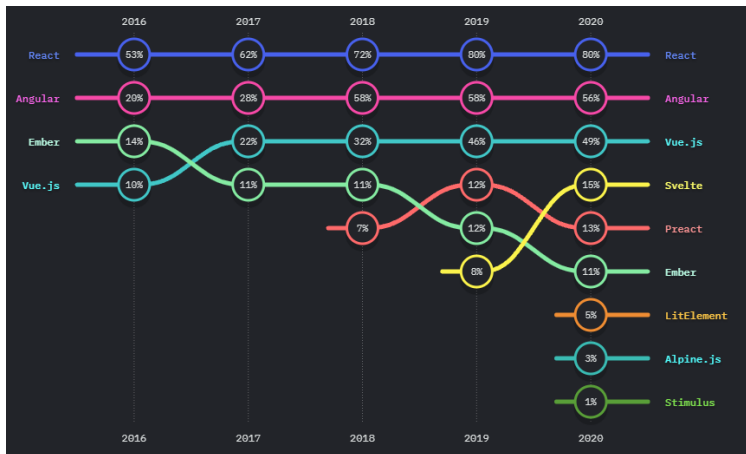


Gambar 3. 2 Grafik trend web framework 2021 (Sumber: stackoverflow.com, 2021)

Hal ini membuktikan bahwa bahasa pemrograman javascript masih sangat dibutuhkan dalam mengembangkan aplikasi, sehingga terdapat beberapa *tech stack* yang dibangun menggunakan bahasa pemrograman tersebut seperti MEAN (Mongodb, Express.js, Angular 13, Node.js), MERN (Mongodb, Express.js, React, Node.js), dan MEVN (Mongodb, Express, Vue.js, Node.js).

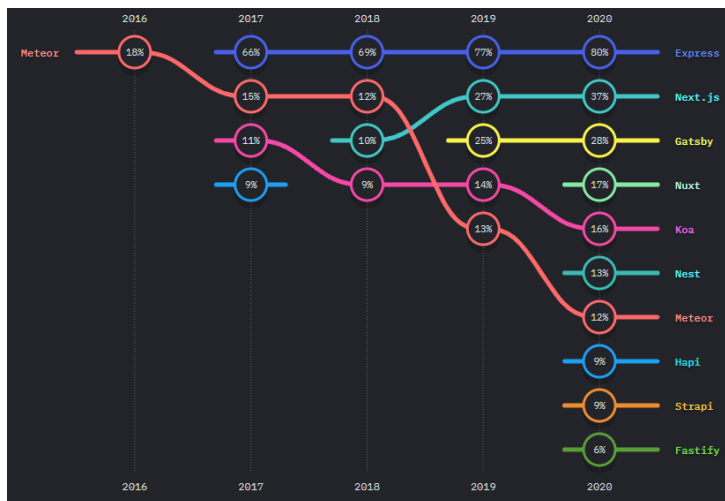
Adapun survey yang dilakukan oleh stateOfJs (Front-end Framework, 2020) penggunaan *frontend framework* yang paling banyak digunakan yaitu React, Angular, dan Vue. React menempati posisi pertama, Angular yang menempati

posisi, dan Vue.js yang menempati posisi ketiga. Survey ini diambil dari tahun 2016 sampai 2020.



Gambar 3. 3 Grafik *trend frontend javascript* berdasarkan *usage* (Sumber: stateofjs.com, 2020)

Untuk *backend framework* yang paling banyak digunakan adalah Express.



Gambar 3. 4 Grafik *trend backend javascript* berdasarkan *usage* (Sumber: stateofjs.com, 2020)

Menjadi pengembang saat ini tidak hanya mencakup pengetahuan yang dibutuhkan untuk pengkodean, tetapi pengembang semakin dipaksa untuk mengikuti perkembangan trend terkini. Dengan semakin banyak perangkat lunak yang direalisasikan dengan JavaScript, baik itu di *frontend* aplikasi maupun di API berbasis Node.js *backend* pekerjaan menjadi lebih kompleks. Ini juga memperumit pengambilan keputusan mengenai teknologi mana yang harus digunakan dalam suatu proyek. *Product lead* sedang dihadapkan dengan ekosistem yang berkembang yang sulit untuk diawasi. Oleh karena itu, diperlukannya beberapa aspek yang akan diukur pada penelitian ini dari beberapa referensi meliputi:

1. Jumlah baris kode(LOC), dan struktur kode (Endra et al., 2021) (Wohlgethan, 2018),
2. Performa (Samra, 2015) (Padmanaba et al., 2020),
3. Menelusuri jumlah fungsi yang dipanggil dalam setiap aksi CRUD (Samra, 2015) (Endra et al., 2021),
4. Penggunaan memori untuk masing-masing *stack* (Samra, 2015),

3.1.2 Pemecahan Masalah

Dari identifikasi masalah yang telah ditemukan maka dibuatlah penelitian analisis komparasi *web development stack* yang berhubungan dengan bahasa pemrograman javascript, mulai dari bagian *frontend* sampai *backend* dalam satu siklus pembuatan sebuah aplikasi. Untuk beberapa macam *tech stack* yang dipilih menjadi objek penelitian ini diantaranya: MEAN, MERN, dan MEVN. Perbedaan dari ketiga *tech stack* tersebut terletak pada huruf ketiga dari setiap akronimnya, yaitu (A) Angular, (R) React.js, dan (V) Vue.js yang merupakan *frontend*

framework berbasis javascript yang digunakan untuk mempermudah pengerjaan tampilan depan pada website. Ketiga *tech stack* lainnya juga memiliki persamaan pada akronim (M) Mongoddb sebagai Database, (E) Express.js sebagai backend web *framework*, dan (N) Node.js sebagai backend *runtime environment*.

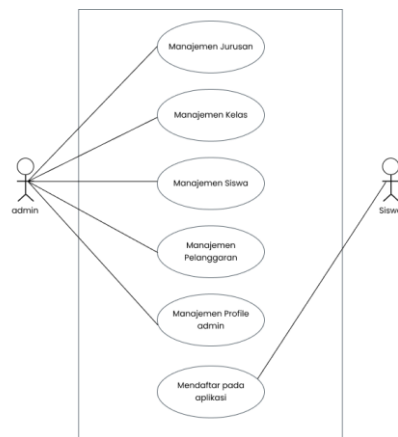
Pada tahap konstruksi aplikasi nantinya akan diuji beberapa aspek meliputi:

1. Jumlah baris kode(LOC) dan struktur kode (Endra et al., 2021) (Wohlgethan, 2018),
2. Performa (Samra, 2015) (Padmanaba et al., 2020),
3. Menelusuri jumlah fungsi yang dipanggil dalam setiap aksi CRUD (Samra, 2015) (Endra et al., 2021),
4. Penggunaan memori untuk masing-masing *stack* (Samra, 2015).

3.2 Perancangan

3.2.1 Gambaran Studi Kasus

Pada penelitian ini sistem akan dibuat seperti *use-case* berikut.



Gambar 3.5 Use case diagram

Commented [K13]: Tidak perlu menyebutkan "Perancangan Sistem". Cukup gunakan "Gambaran Studi Kasus" Dan tekankan bahwa aplikasi ini nantinya akan dibuat dalam 3 versi dengan stack yang berbeda

Commented [KWP14R13]: OK

Nantinya sistem akan dibuat identik dalam tiga versi *stack* yang berbeda (Angular, React, Vue) didalamnya terdapat beberapa menu antara lain: menu kelas, jurusan, siswa, pelanggaran, dan profil admin.

3.2.2 Perancangan Data

Perancangan data dibuat menggunakan bantuan JSON karena database yang digunakan adalah MongoDB yang termasuk database NoSQL yang dimana memiliki skema yang fleksibel dan relasinya berbentuk *embedded object* ataupun *reference*. Bentuk dokumen dari NoSQL adalah JSON yang mudah dibaca dan dimengerti.

1. Siswa.json

```
_id: ObjectId("62d264d7376c191f93230770")
namaLengkap: "Aldo farros "
NIS: "2203020002"
NISN: "9998575202"
tempatLahir: "Malang"
tanggalLahir: 1999-05-05T00:00:00.000+00:00
kewarganegaraan: "WNI"
role: "siswa"
status: "y"
> akademikSiswa: Object
> orangTuaWali: Object
> kesehatanSiswa: Object
kelas: ObjectId("624dc2faae67093062b895fe")
user: ObjectId("62d25f67376c191f932306ea")
createdAt: 2022-07-16T07:12:23.833+00:00
updatedAt: 2022-07-16T07:12:23.833+00:00
__v: 0
```

Gambar 3. 6 Dataset *siswa*

2. Jurusan.json

```
_id: ObjectId("61ae09fc1513eea7986db99d")
bidangKeahlian: "TEKNOLOGI DAN REKAYASA"
programKeahlian: "TEKNIK ELEKTRONIKA"
paketKeahlian: "Teknik Mekatronika"
singkatan: "Metro"
foto: "1638795772004-metro.png"
status: "y"
createdAt: 2021-12-06T13:02:52.050+00:00
updatedAt: 2022-04-02T14:31:21.004+00:00
__v: 0
kode: "03"
warna: "#3f50e3"
```

Gambar 3. 7 Dataset jurusan

3. Kelas.json

```

_id: ObjectId("624d9d8c868e6272e89a4be1")
kode: "22100301"
tingkatan: "x"
keahlian: ObjectId("61ae09fc1513eea7986db99d")
abjad: "a"
tahunAjaran: "2022"
keterangan: ""
status: "y"
createdAt: 2022-04-06T14:02:52.185+00:00
updatedAt: 2022-07-16T07:17:44.038+00:00
__v: 0
jumlahSiswa: 2

```

Gambar 3. 8 Dataset kelas

4. Users.json

```

_id: ObjectId("61b9bf597090bdcffe336cae")
email: "admin@gmail.com"
password: "$2a$10$ZDYHz1aAvdPVI8QJesIjV0dcvdIHQ5sWxGcJzrn08o6bo7Yt7C670"
role: "admin"
status: "Y"
namaLengkap: "Bima Reynaldi S"
NISN: "9998575190"
tempatLahir: "Malang"
tanggalLahir: 1999-02-02T00:00:00.000+00:00
namaIbu: "Ipuz"
telp: "081330982004"
createdAt: 2021-12-15T10:11:40.466+00:00
updatedAt: 2022-07-18T00:46:20.119+00:00
__v: 0

```

Gambar 3. 9 Dataset user

5. Pelanggaran.json

```

_id: ObjectId("61b501c919f43e80ff64e6fe")
kategori: "kelakuan"
jenisPelanggaran: "Membawa atau mengonsumsi miras, ganja, narkoba, dan sejenisnya di seko..."
jumlahPoin: 100
createdAt: 2021-12-11T19:53:45.327+00:00
updatedAt: 2022-04-18T06:27:22.519+00:00
__v: 0

```

Gambar 3. 10 Dataset pelanggaran

3.3 Rancangan Pengujian

Metode pada rancangan pengujian yang akan dilakukan terhadap analisis komparasi web development stack berbasis javascript pada fase konstruksi aplikasi: MEAN, MERN, dan MEVN meliputi:

3.3.1 Software Metrics

Software metrics atau metrik perangkat lunak adalah ukuran karakteristik perangkat lunak yang dapat diukur atau dihitung. Metrik perangkat lunak menjadi penting karena menawarkan cara cepat untuk melacak pengembangan perangkat lunak, menetapkan tujuan, dan mengukur kinerja. Tetapi penyederhanaan pengembangan perangkat lunak dapat mengganggu pengembang perangkat lunak dari tujuan untuk memberikan perangkat lunak yang bermanfaat dan meningkatkan kepuasan pelanggan, tim pengembangan perangkat lunak mungkin menganggap lebih penting untuk benar-benar melakukan pekerjaan daripada mengukurnya. Sehingga sebaiknya mengukur dan menganalisis tidak harus memberatkan atau sesuatu yang menghalangi pembuatan kode.

3.3.1.1 Line of Code (LOC)

Line of Code atau baris kode adalah ukuran umum pengembangan perangkat lunak. Terdapat dua cara untuk menghitung setiap baris kode:

1. Salah satunya adalah menghitung setiap garis fisik yang diakhiri dengan pengembalian atau singkatnya menghitung satu buah fungsi dianggap sebagai satu baris kode.
2. Setiap pernyataan logis dapat dianggap sebagai baris kode.

Commented [K15]: Pastikan sudah menuliskan sumber referensi tentang alasan penggunaan metric-metric tersebut.

Commented [KWP16R15]: OK

3.3.1.2 Performance

Performance dari sebuah website diukur dari seberapa efisien kodenya, seberapa optimal arsitekturnya, apakah sistem mampu menskalakan, memuat waktu halaman, atau fungsionalitas utama. Beberapa metric yang dapat diukur antara lain:

1. *First Contentful Paint*: Mengukur berapa lama waktu yang dibutuhkan browser untuk merender bagian pertama konten DOM setelah pengguna menavigasi ke halaman (Performance-scoring, 2021).
2. *Speed Index*: Mengukur seberapa cepat konten ditampilkan secara visual selama pemuatan halaman (Performance-scoring, 2021).
3. *Largest Contentful Paint*: melaporkan waktu render dari gambar atau blok teks terbesar yang terlihat dalam area pandang, relatif terhadap saat halaman pertama kali mulai dimuat (Performance-scoring, 2021).
4. *Time to Interactive*: mengukur berapa lama waktu yang dibutuhkan halaman untuk menjadi sepenuhnya interaktif (Performance-scoring, 2021).
5. *Total Blocking Time*: mengukur jumlah total waktu halaman diblokir untuk merespons input pengguna, seperti klik mouse, ketukan layar, atau penekanan keyboard. Jumlahnya dihitung dengan menambahkan bagian pemblokiran dari semua tugas panjang antara *First Contentful Paint* dan *Time to Interactive* (Performance-scoring, 2021).
6. *Cumulative Layout Shift*: ukuran ledakan skor pergeseran tata letak terbesar untuk setiap perubahan tata letak tak terduga yang terjadi selama seluruh umur halaman (Performance-scoring, 2021).