

# Tech Stack

**Silver** - Chapter 0 - Topic 4

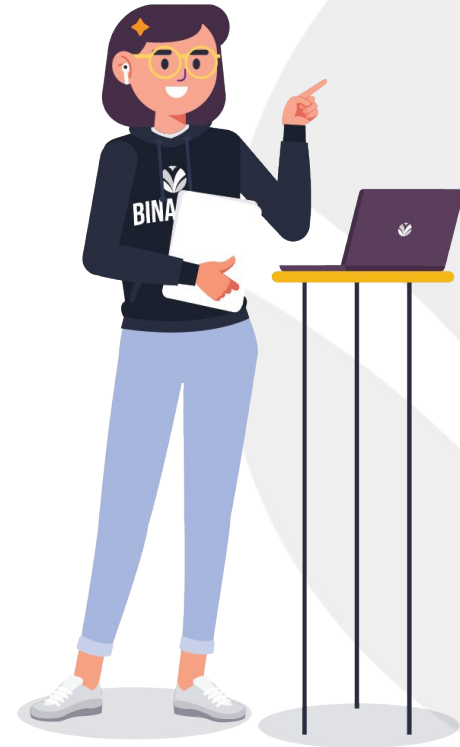
Selamat datang di **Chapter 0 Topic 4**  
online course dari  
Binar Academy!



## Welcome back!

Karena sudah dibahas di topic sebelumnya, kamu kan udah paham sama tugas dari frontend engineer dan backend engineer. Nah, di topik ini kita akan **membedah tech stack dan tools yang biasa dipakai buat menjalankan tugas mereka.**

Kalau gitu, yuk langsung aja kita kepoin~



## Apa yang bakal kamu pelajari di Topic 4 ini?

Pada kesempatan kali ini, kamu akan mengenal lebih dalam mengenai:

- 1) Mengetahui apa itu tech stack dan tools
- 2) Mengetahui berbagai macam tech stack
- 3) Mengetahui berbagai macam jenis tools



## Mari kita masuk ke pembahasan materi

Oke, sekarang kita bayangkan ada di sebuah pabrik mobil:

- Product owner adalah pemilik pabrik
- Engineer adalah teknisi, dan mobil adalah produk aplikasi yang akan dibuat.

**Maka, tech stack ini adalah metode yang digunakan untuk membuat mobil sementara tools adalah peralatan yang membantu menjalankan metode.**



### Tech stack yang wajib dikuasai oleh seorang frontend engineer sendiri itu ada tiga, yakni HTML, CSS, dan JavaScript

Karena sering berkuat sama tampilan depan dari sebuah aplikasi, tech stack yang dipakai harus bisa membantu frontend engineer buat bikin desain, konten, gambar, navigasi, sampe link yang berkaitan.

Cus, langsung kita bahas satu-satu mulai dari HTML!



Kalau aplikasi diibaratkan mobil, HTML ini adalah kerangka yang nyusun body mobil yang nantinya dipake buat nyusun bagian lainnya kaya kap, bumper, pintu, sampai roda.



`(<header>, <link>, <p>)`

**HTML** itu singkatan dari **Hypertext Markup Language** yang bisa digunakan sebagai "alas" untuk membuat dan menyusun bagian paragraf, heading, dan link menggunakan **tag** kaya contoh gambar di samping pada halaman depan aplikasi.

Gampangnya, HTML itu kerangka mobil sementara paragraf, heading, dan linknya itu kap, bumper, dan pintu mobil.



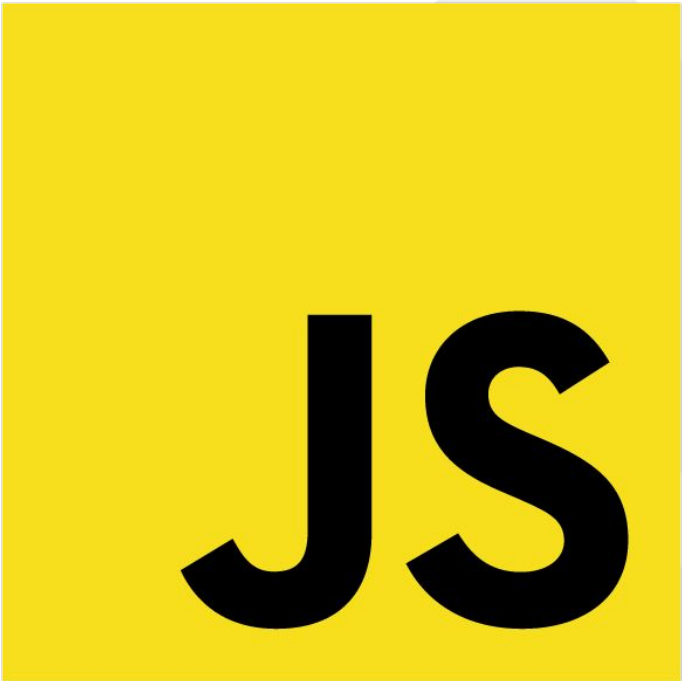
Kalo tadi HTML itu kerangka mobil, maka CSS adalah cat dan coating yang bikin eksterior mobil makin ciamik.

**CSS sendiri singkatan dari Cascading Styling Sheet yang berfungsi buat bikin tampilan aplikasi yang menarik** mulai dari tata letak, pemilihan warna, sampai ukuran huruf yang pas biar enak dipandang.



Sekarang body mobil udah jadi, udah dicat, dan dikasih coating yang ciamik. Biar pintu mobil bisa dibuka, jendela bisa diturunin, dan pintu bagasi bisa dinaikkan, kita butuh sistem yang ngatur itu semua. Sistem ini yang disebut dengan JavaScript.

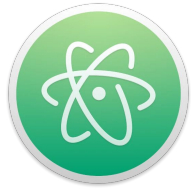
JavaScript ini adalah bahasa pemrograman intrepreter yang berfungsi agar tampilan depan aplikasi lebih interaktif, dinamis dan pastinya bisa dipake sama penggunanya.

A large yellow square containing the letters 'JS' in a bold, black, sans-serif font, representing JavaScript.



Buat menyatukan tiap kerangka, mengecat, sampe bikin sistem kunci mobil, kita pasti butuh peralatan nih. Peralatan ini yang tadi disebut sebagai tools.

Tools yang biasa dipake sama frontend engineer dibagi lagi jadi tiga, yaitu tools yang dipake pas bikin teksnya, pas ngecek hasil teksnya, sama buat ngontrol kalo ada masalah pada teksnya di kemudian hari.



ATOM



Sublime Text



Visual Studio Code

**Tools yang dipake buat nulis teks dari tech stack disebut dengan text editor atau IDE (Integrated Development Environment).**

Text editor yang biasa dipake sama Frontend engineer ada Atom, Sublime, dan Visual Studio Code.



Kalo teksnya udah ditulis, kita butuh tools yang bisa menampilkan teks tersebut biar kelihatan hasilnya kaya gimana. Tools ini yang disebut sebagai browser development tools.

Browser development ini biasanya pake browser yang biasa kamu pake buat internetan kaya Mozilla Firefox dan Google Chrome.



# git

Terakhir, kita butuh tools yang disebut **version control (Git)** yang tugasnya buat membantu nge-track kalau ada perubahan dalam kodingan kita.

Oke, khusus buat version control ini bakal dibahas lebih lanjut di chapter selanjutnya. Kamu harap bersabar ya!

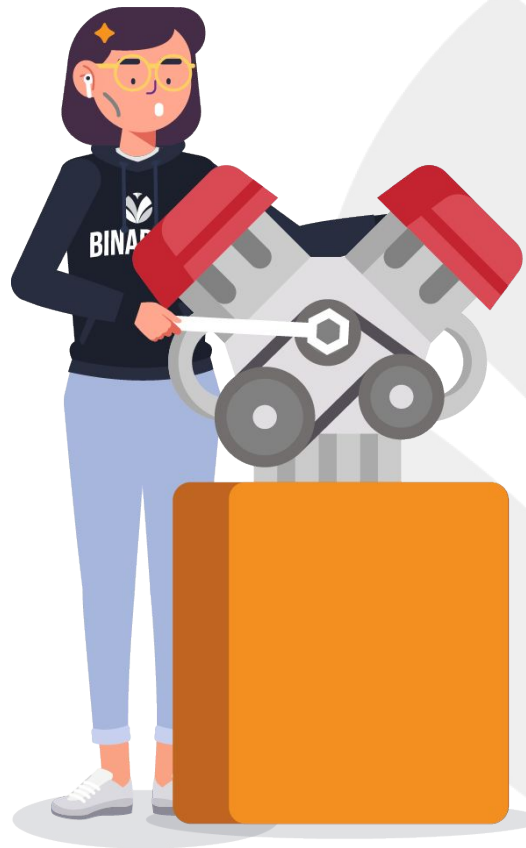


### Mobil kita udah jadi nih tapi mobilnya masih belum bisa jalan 😞

Biar mobilnya bisa jalan, kita butuh satu sistem mesin. Nah, kalo di pabrik mobil tadi, teknisi bertugas buat bikin sistem mesin ini adalah para backend engineer.

**Simplenya, kalo ga ada backend engineer, aplikasimu gak bakal jalan.**

Biar aplikasimu bisa jalan, kamu butuh asupan data nih. Buat menyediakan data ini, backend engineer harus bisa ngutak-ngatik tech stack yang ada.



**Tech stack yang dipake sama backend engineer ini biasanya dikelompokkan jadi 4 golongan tergantung fungsinya masing-masing**

- 1) Web server
- 2) Web framework
- 3) ORM (Object Relation Mapping)
- 4) Database: SQL (Structured Query Language)

Kita bahas satu-satu yuk!





Web server ini gampangnya adalah kurir yang menyalurkan permintaan dari browser kaya mozilla tadi ke sumber data yang dibutuhkan. Kalau udah dapet data yang diminta, dia juga yang bakal mengantarkan data ini kembali ke browser buat ditampilin ke pengguna.

Web server yang biasanya dipakai sih ada Apache, Nginx, sama Microsoft IIS (Internet Information Service).

The Nginx logo, consisting of the word "NGINX" in a bold, green, sans-serif font.



Sedangkan web framework gunanya buat ngatur berbagai fungsi sama konsep biar membentuk satu sistem tertentu. Tujuannya biar aplikasi yang dibuat terstruktur dengan rapi.

Pilihan web framework ini sebenarnya banyak banget tapi yang sering dipake tuh ada Express by JavaScript, Django by Python, Laravel by PHP, dan Ruby on Rails by Ruby.



Sama kayak namanya, database berupa kumpulan data yang disimpan secara sistemik di dalam sistem komputer. **Database inilah yang jadi sumber data buat para pengguna aplikasi.**

Berdasarkan tipe bahasa yang digunakan, database dibagi menjadi 2 jenis yaitu tipe SQL dan NoSQL. Database tipe SQL contohnya MySQL, PostgreSQL, dan MariaDB. Sementara itu database tipe NoSQL contohnya MongoDB.

**Adapun relasi untuk menghubungkan antara web framework dan database disebut ORM.**

**Fiuuhh..akhirnya setelah melalui proses panjang, mobil yang kita pengen udah jadi! 🌟😊**

Hmm.. tapi kalau dilihat-lihat, mobilnya bakal lebih bagus kalau velg-nya diganti terus ditambahin speaker biar bisa karaoke di mobil. Kayaknya bakal bagus deh kalo dimodifikasi. 🤔



Kalo kamu ingat, di topik 3 kita sempat bahas tentang mobile engineer. Oke, biar sama-sama inget, kita bahas lagi ya fungsi si mobile engineer ini.

Kalau di dunia nyata, modif mobil tuh prinsipnya gak bisa sembarangan. Nah, modif itu gak cuma ngubah bentuk luarnya aja tapi juga harus ada penyesuaian biar gak ganggu fungsi-fungsinya.

Terus, modif mobil ini biasanya butuh teknisi khusus yang paham caranya modif biar gak konslet. Kalau di pembuatan aplikasi, tugas teknisi ini diisi sama mobile engineer.



## Meskipun sama-sama fokus ke tampilan depan web, frontend engineer sama mobile engineer itu beda

Kalau frontend engineer tugasnya ngurusin tampilan depan web, Mobile engineer ini tugasnya lebih spesifik.

Dia fokus buat bangun tampilan web biar ramah buat pengguna aplikasi smartphone. Mobile engineer ini bakal dibutuhin banget pas produk udah berada di tahap production environment alias aplikasinya udah masuk di Play Store atau App Store.

### Mobile Engineer



### Frontend Engineer



### Ini nih, daftar tech stack yang biasa dipakai sama Mobile engineer!

- PWA (Progressive Web App)
- Native
- Hybrid
- JavaScript Native
- Cross-compiled



PWA (Progressive Web App) ini tuh jadi metode yang paling sederhana karena cuma bikin tampilan website jadi mirip sama tampilan sebuah aplikasi.

Kamu pernah nggak mengunjungi satu website dari browser di smartphone kalian tapi isi websitenya interaktif banget dan berasa kaya lagi pake aplikasi? Nah, metode itu yang disebut sama PWA.

Sedangkan Native, doi ini sistemnya misahin metode buat bikin mobile apps di sistem operasi Android dan iOS.





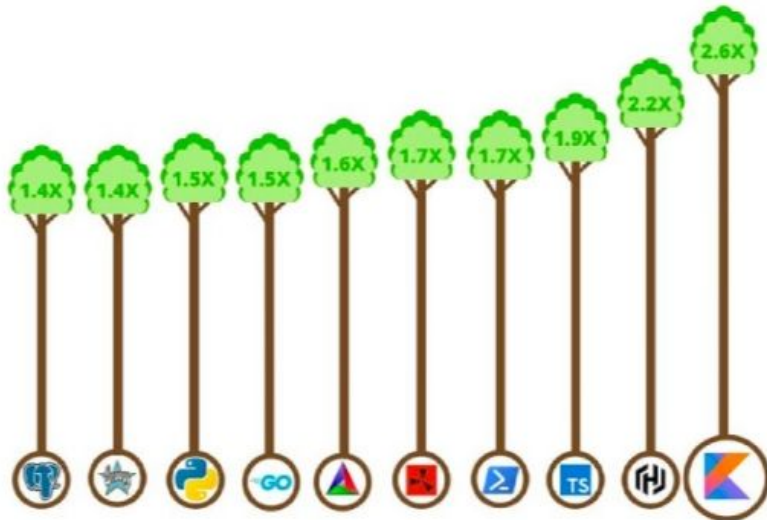


Spesialisasi buat mobile engineer ini kebanyakan dibagi dua, spesialis Android dan spesialis iOS.

Kenapa dibagi? Karena bahasa dan tech stack yang dipake sama keduanya beda alias native. **Kalo Android pake Kotlin, iOS pake Swift.** Eitsss! Tapi buat materi bahasa ini kita save di topic selanjutnya ya! 🤗



### Pertumbuhan bahasa pemrograman open-source tercepat



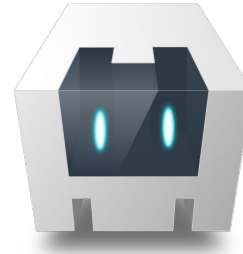
**Sekarang ini, Android udah jadi yang paling banyak di dunia jauh ngalahin iOS**

Karena Android ini biasanya pake bahasa Kotlin, otomatis tren penggunaan bahasa ini juga ikutan naik.



Lanjut! Yang ketiga ada Hybrid. Sama kaya namanya,  
**Hybrid ini campuran dari dua tech stack antara PWA  
sama Native.**

Kalau tools yang biasa dipake di metode Hybrid ini ada  
Apache Cordova atau Ionic's Capacitor.



APACHE  
CORDOVA™



**Capacitor**

Oke, lanjut ke JavaScript Native. Tech stack yang ini bisa dibilang yang paling universal karena kalau kita pake metode ini, kita bisa bikin mobile apps yang bisa dipakai di sistem operasi Android dan iOS sekaligus.

Framework yang biasa dipake tuh ada Native Script atau React Native. Tools buat framework ini ada Visual Studio Code dan Android Studio.



Visual Studio Code



Android Studio

Terakhir nih, ada tech stack yang namanya cross-compiled.

Kalau tadi PWA cuma bikin website jadi punya tampilan kaya aplikasi, **cross-compiled ini yang ngubah website jadi aplikasi beneran**. Salah satu contoh dari cross-compiled ini ada Xamarin yang dikeluarkan sama Microsoft.



**Finally! Kita udah kelar nih bahas materi. Jadi apa aja yang udah kamu pelajari di topik tech stack ini? Let's wrap it out!**

- 1) Definisi tech stack dan tools
- 2) Berbagai jenis tech stack buat Frontend engineer
- 3) Berbagai jenis tools buat Frontend engineer



- 4) Berbagai jenis tech stack buat Backend engineer
- 5) Berbagai jenis tools buat Backend engineer
- 7) Perbedaan Frontend engineer dengan Mobile engineer
- 8) Berbagai jenis tech stack buat Mobile engineer



Yuhuuu! Kamu udah berhasil  
menamatkan **Chapter 0 Topic 4** 🇺🇸🇮🇩

Nanti, di terakhir alias topic 5 kita bakal  
ngebahas tentang Bahasa Pemrograman.  
Udah siap?

