



`</>_1`

Hello World

Enter The Real
World

Andaru Triadi

The Real World

2021 – 2022

Hello World

Silahkan baca e-book ini bersama secangkir kopi hangat atau dingin. Senyaman kamu :). Ready to enter the real world with me? Lessss Go.....

DAFTAR ISI

DAFTAR ISI	iv
KENALAN YUK!.....	1
# Intro	2
# Why Programmer?.....	3
# Dunia Perkuliahan	6
# Freelance	7
# Pengalaman Kerja.....	9
# Komunikasi Itu Penting	10
# Investasi Ilmu.....	11
BELAJAR NGODING.....	12
# Push Rank.....	13
# Teknik Pomodoro	15
# Einshower Matrix	17
# Programmer Yang Tersesat.....	20
# Seni Merasa Cukup.....	22
# Belajar Bahasa Inggris	24
PROGRAMMING.....	24
# Memilih Framework	25
# Nikmatnya Error	27
# Dokumentasi Adalah Sahabat Mu!	29
# Fundamental	31

# DRY (Don't Repeat Yourself)	33
# API (Application Programming Interface)	35
# Relasi Database	37
# ORM (Object Relational Mapping)	38
# Baru Kenal Livewire	39
STIGMA.....	42
# Si Paling Bisa	43
# Introvert	45
# Jomblo	47
# Kutu Buku	49
# Kurang Pergaulan	50
ADVICE.....	48
# Meditasi	49
# Keluar Sejenak	51
# Jangan FOMO	53
# Puasa Sosial Media	55
THANK'S TO.....	58

KENALAN YUK!

Intro



Hai, nama saya Andaru Triadi. Anak ke-tiga dari tiga bersaudara. Lahir di Ponorogo, Jawa Timur. Lulusan Animasi yang melanjutkan studinya ke dunia IT (Coding).

Sebuah takdir membawa saya untuk memperdalam dunia IT. Dunia IT tidak seperti apa yang kita lihat di film. Memperdalam dunia IT menurut saya sangat menantang.

Why Programmer?



Sumber : unsplash

Ketika smp kelas 1, saya mengalami patah tulang karena tertabrak motor. Takdir baik ini mengarahkan saya kepada hal baru, yaitu komputer. Sebelumnya sangat menyukai futsal, setelah patah jadi lebih menyukai komputer.

Berjalannya waktu, ketika smk saya terobsesi menjadi hacker. Mempelajari apapun ilmu yang terkait hacker. Mendekati kelulusan, saya menyadari bahwa di dunia hacking saya jelas belum bisa

memastikan kedepannya akan baik atau buruk terhadap diri saya pribadi.

At the time, di salah satu kafe tebet, saya melihat seseorang sedang membuat program dengan macbooknya. Apapun yang terlihat dimata sulit, pasti akan saya coba (suka rintangan).

Saya mulai sedikit tertarik akan dunia programming ketika kelas 3 smk, Berawal dari guru yang mengajak saya untuk mengikuti Lomba LKS.

Kenapa memilih menjadi programmer?

Dengan menjadi programmer saya merasa dapat bermanfaat bagi orang lain. Contoh membuat aplikasi absensi, dengan adanya absensi online, mereka tidak perlu membuat daftar absensi secara manual di kertas maupun excel. Hal ini mempermudah pengguna maupun pengurus.

Menjadi programmer bukan hanya membuat aplikasi saja, tetapi belajar memahami kebutuhan orang lain dan belajar menyelesaikan sebuah masalah (Problem Solving).

Semakin mengasah skill programming, maka secara otomatis akan menambah kreatifitas dalam membuat produk.

Disatu sisi, programmer dapat bekerja secara offline maupun online. Sangat fleksible. Umumnya programmer yang memiliki waktu luang di tengah kesibukan pekerjaan, mereka dengan mudah melakukan freelance.

Dunia Perkuliahan



Sumber : republika.co.id

Awalnya saya berniat untuk bekerja, Namun takdir berkata lain. Saya memutuskan untuk kuliah di Universitas Bina Sarana Informatika atau biasa dikenal dengan kalimat “Kuliah? BSI aja”.

Masuk Universitas BSI di tahun 2019.

Freelance



Sumber : Unsplash

Pertama kali freelance, saya mendapatkan project pembuatan web berbasis PHP dari salah satu guru saya pada saat SMK. Saat itu saya membuat struktur database dengan MySQL.

Tahun demi tahun freelance berjalan dengan lancar sampai di semester 7 saya mencoba untuk fokus mencari pekerjaan karena sudah jarang mendapatkan project.

Freelance tidak seperti yang kamu lihat di youtube, setelah freelance beli komputer dengan harga mahal. Saya rasa mereka pun tidak murni dari pendapatan freelance, melainkan pendapatan dari youtube juga.

Bila job sedang kosong, terkadang saya mempelajari skill baru diluar programming. Paling sering belajar programming yang terkait backend.

Hidup naik turun, sama seperti freelance. Terkadang banyak job, terkadang kosong job. Sudah sewajarnya. Tidak perlu risau, cukup jalankan dan syukuri.

Pengalaman Kerja



Sumber: Pexels

May 2021, saya mendapatkan tawaran pekerjaan di salah satu biro jasa yang ada di Jakarta. Finally saya mendapatkan pekerjaan pertama sebagai Web Developer! Pada saat itu stack yang saya gunakan Laravel, Livewire, dan MySQL.

Disisi lain dari percodingan, saya dan owner mengalami miskomunikasi dimana dampak dari hal itu membuat saya dipecat.

Komunikasi Itu Penting



Sumber : Pexels

Kurang nya komunikasi antara programmer dengan client membuat kerugian waktu untuk kami. Namun, hal yang kami pelajari kali ini bahwa programmer tidak hanya duduk didepan komputer dan ngoding, melainkan programmer pun wajib untuk memiliki komunikasi yang baik. Baik berkomunikasi dengan client, atasan, maupun level yang setara dengannya.

Investasi Ilmu



Sumber : Pexels

Sebanyak apapun uang yang dimiliki manusia, akan habis pada waktunya. Betul? Jika jawabannya tidak, mungkin kamu bisa bertahan dengan uang mu yang banyak hingga bertahun-tahun.

Bagaimana kalau kamu sudah memiliki keluarga? bahkan anak?

Saya dulu ketika dapat project, tapi hasilnya tidak ditabung. Alhasil ketika butuh upgrade skill tidak bisa. Teringat nasihat orang tua pada zamannya “Tuntutlah Ilmu sampai negeri china”

Nasihat klasik ini kita semua tahu, tapi belum tentu semua memahami atau menerapkan, karena beberapa hal. Pada akhirnya saya mengumpulkan uang untuk membeli course dan mengikuti bootcamp.

Investasi Ilmu sama seperti Investasi Saham, return (hasil) yang akan kita dapat bukan sekarang, besok, maupun bulan depan, namun tahun yang akan datang. Semua ilmu akan sangat bermanfaat bila kita konsisten mempelajarinya.

BELAJAR NGODING

“BUKAN KAMU YANG TIDAK BISA, MUNGKIN CARA
KAMU BELAJAR YANG KURANG TEPAT!”

Push Rank



Sumber : entrepreneur.com

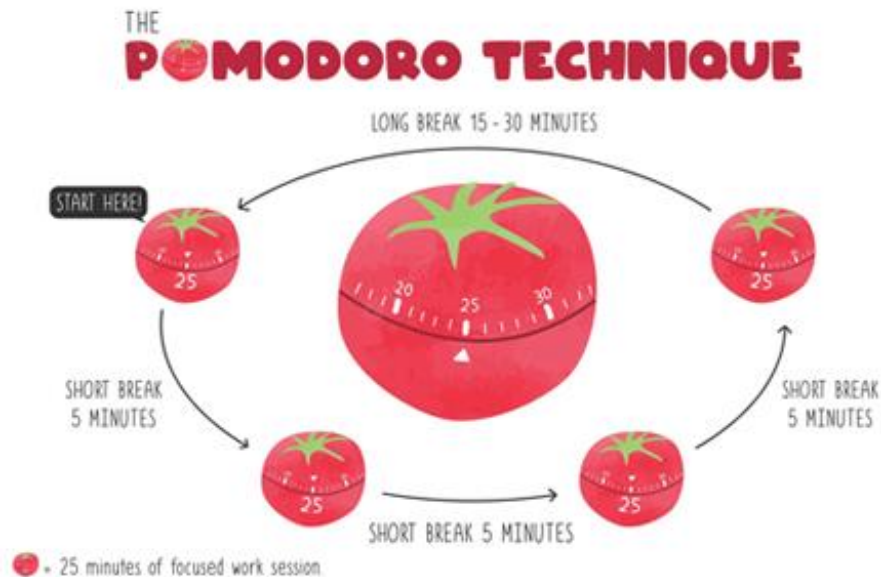
Tidak asing mendengar push rank dizaman sekarang. Yang dimaksud disini bukan push rank Mobile Legend, push rank yang dimaksud adalah kamu belajar coding secara terus menerus / berlebihan sehingga kamu pusing.

Belajar coding tidak perlu di push seminggu selesai, tanpa kamu belajar pun sudah tertinggal, karena dunia IT terus berkembang secara pesat.

Pertama kali saya belajar coding, saya push 1 minggu, alhasil tidak paham. Pada saat itu saya belajar html dan css. Belajar coding tidak perlu terburu-buru.

Kita hanya perlu untuk konsisten dalam belajar dalam beberapa jam. Hindari belajar coding seharian, karena hal itu dapat membuat mu tidak produktif. Jangan lupa untuk ber-instirahat.

Teknik Pomodoro



Sumber : Freepik

Sudah belajar coding beberapa jam tapi tidak memahami apa yang sudah kamu pelajari? Kita satu nasib!

Menurut PSB Academy, sains telah menunjukkan bahwa belajar yang paling efektif adalah antara jam 10 pagi hingga jam 2 siang, dan jam 4 sore hingga jam 10 malam.

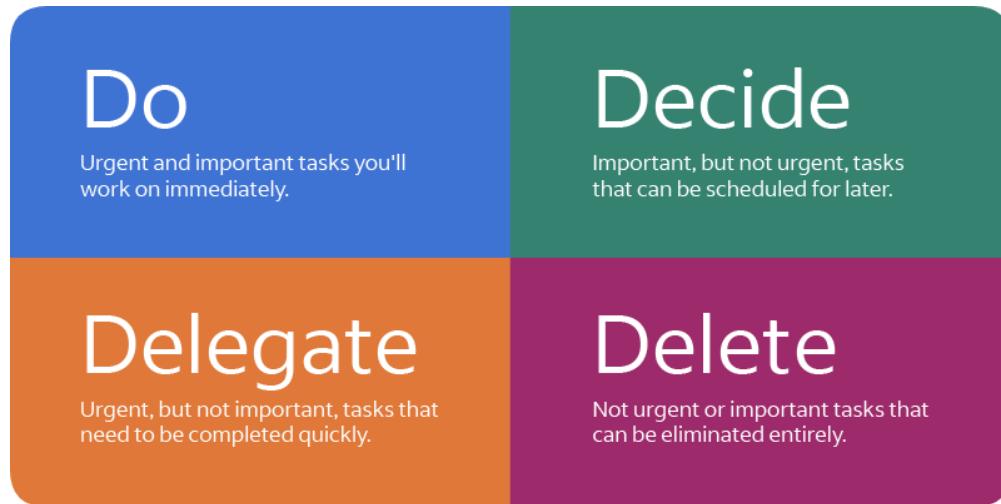
Ketika smk, saya mempelajari bootstrap 4, namun tidak kunjung mengerti. Iseng-iseng mencari tahu kenapa sih sudah lihat tutorial, baca buku tapi tidak juga paham? Apa saya tidak cocok untuk menjadi programmer?

Pada bulan yang sama, saya menemukan teknik belajar yang bernama Pomodoro. Pomodoro dikembangkan oleh mahasiswa bernama Francesco Cirillo.

Dengan Pomodoro, kita memecahkan waktu pembelajaran menjadi beberapa bagian. 1 pomodoro berisi 25 menit dan 5 menit istirahat. Jika kamu memiliki target 3 jam belajar coding, kamu dapat memecahkan menjadi 6 bagian.

Setelah beberapa bulan kemudian saya mengerti bahwa bukan saya tidak bisa, tapi cara belajar saya yang salah. Salah satu kesalahan saya dalam belajar, yaitu 1x belajar 1-3 jam tanpa istirahat.

Einshower Matrix



Sumber : accurate.id

Penyakit yang biasa diderita freelancer, tidak jauh dari management waktu dan management task. Di awal freelance, saya tidak tahu bagaimana cara nya mengatur mana dulu task yang harus saya kerjakan.

Kita bisa membagi task menjadi 4 kuadran, kuadran pertama : penting dan mendesak, kuadran kedua : penting tapi tidak mendesak, kuadran ketiga : tidak penting tapi mendesak, kuadran keempat : tidak penting dan tidak mendesak.

Mengerjakan task dimulai dengan kuadran pertama (Do). Jadwalkan bila task penting tapi tidak mendesak (schedule). Delegasikan bila task tidak penting tapi mendesak (delegate). Hilangkan bila task tidak penting dan tidak mendesak (delete).

Bagaimana cara kita tahu bahwa sebuah task sangat penting? Dalam penerapan awal einhower matrix, saya pun bingung bagaimana cara mengetahui sebuah task sangat penting.

Namun agak sedikit tricky ya, saya men-sortirnya seperti ini. Berikut cara saya mengimplementasikan einhower matrix pada kegiatan kampus.

1. Jika suatu task sangat penting namun memiliki deadline dalam 1 minggu, letakkan di kuadran Do. Seperti: dalam 3 hari kedepan, tugas web programming I harus dikumpulkan di laman web kampus.

2. Jika suatu task sangat penting namun memiliki deadline lebih dari 1 minggu, letakkan di kuadran Schedule. Seperti : dalam 3 minggu kedepan diharuskan untuk mengirim video presentasi.
3. Jika suatu task tidak penting namun mendesak, letakkan pada kuadran schedule. Seperti disuruh orang tua beli sambal, namun kamu sedang sibuk. Kamu dapat menyuruh adik mu untuk membeli. Itu yang namanya delegasi.
4. Jika suatu task tidak penting dan juga tidak mendesak, letakkan pada kuadran delete. Seperti : kebiasaan bermain game disela mengerjakan tugas hingga lupa waktu dan prokrastinasi melanda...

Programmer Yang Tersesat



Sumber : grid.id

Saya mengalami ini ketika duduk di bangku sekolah. Saya mempelajari framework codeigniter tanpa mempelajari bahasa pemrograman php. Sedangkan codeigniter dibuat menggunakan php.

Saya merasa Bingung dan Tersesat.

Saat itu saya belum kenal dengan yang namanya **roadmap**. Setelah mengenal roadmap, belajar coding menjadi lebih tertata rapih. Mulai dari html, css, javascript, hingga framework lainnya.

Setiap Roadmap berbeda-beda tergantung kamu ingin terjun kemana. Roadmap biasanya ada beberapa, yang terkenal itu Roadmap Frontend Developer atau Roadmap Backend Developer. Akan dibahas pada lain waktu.

Bila memiliki senior atau mentor disarankan untuk meminta bimbingan sehingga proses menjadi lebih mudah dan terstruktur.

Otodidak tanpa roadmap atau mentor dapat merugikan waktu belajar mu. Bukan berarti otodidak itu buruk ya...

Seni Merasa Cukup



Sumber: istock

Setelah berhasil mempraktekkan teknik pomodoro dan einshower matrix, rasanya saya merasa tidak puas meskipun sudah menyelesaikan todolist hari ini.

Merasa tidak puas dengan skill yang sudah dimiliki salah satu penghalang masuknya ilmu ke diri kita. Ketika saya menerapkan seni merasa cukup, berapa pun todolist yang saya selesaikan akan tetap merasa cukup.

Kata Marcus Aurelius “Yang diberikan alam tidak lah banyak, namun manusia nya lah yang tidak merasa cukup.”

Marcus Aurelius adalah sosok filsafat yang sangat saya kagumi. Setiap hari saya membuat todolist harian dan men-set apa saja yang ingin dipelajari untuk hari ini.

Mengasah perasaan cukup membuat berapapun progress belajar kamu menjadi lebih baik. Kamu tidak merasa dikejar-kejar lagi oleh rasa ketidakpuasan mu.

Ada banyak orang yang sudah kaya raya namun masih tersiksa karena merasa tidak puas akan hartanya, pencapaiannya, jabatannya.

Dalam implementasi nya, saya belajar melalui youtube mas Adjie Santosoputro.

Belajar Bahasa Inggris



Sumber : leverageedu.com

Mulai dari dokumentasi hingga pesan error menggunakan bahasa inggris. Tidak perlu belajar bahasa inggris secara advance(mahir), yang penting dapat memahami arti teks berbahasa inggris.

Ketika smp, saya mengikuti les bahasa Inggris di Jakarta Timur. Awalnya yg saya tahu bahasa inggris untuk menghadapi UN.

Masuk dunia coding, saya baru paham sepenting ini bahasa inggris. Umumnya bahasa

pemrograman atau framework lainnya dibuat oleh orang luar negeri. Kan ada google terjemahan?

Iya betul, tapi jika kamu mengerti lebih bagus lagi. Jadi lebih menghemat waktu, tidak mondar mandir ke google terjemahan. Semakin sering kamu membaca dokumentasi semakin banyak kosakata (vocabularies) yang kamu miliki.

Selain itu, kamu dapat berkomunikasi dan berbaur di komunitas luar negeri. Dan banyak manfaat lainnya.

PROGRAMMING

“TENTANG PROGRAMMING DAN KITA!”

Memilih Framework



Hal pertama yang membingungkan menurut saya adalah pemilihan framework. Saya tidak tahu saya interest kemana. Backend kah? atau Frontend kah?

Setelah mempelajari frontend basic dan backend basic, saya memutuskan untuk memperdalam backend. Pilihan pertama saya codeigniter. Setelah 3 tahun menggunakan codeigniter, saya memutuskan untuk pindah ke Laravel.

Apakah codeigniter jelek?

Tidak, semua framework ada kelebihan dan kekurangan masing". Codeigniter lebih cocok bila digunakan untuk membuat project kecil. Sedangkan Laravel sebaliknya. Apapun yang kamu pilih, itu yang terbaik.

Programmer terbaik bukan mereka yang bisa banyak framework, tapi mereka yang mahir di beberapa framework dan bisa menyesuaikan kebutuhan client/perusahaan nya.

Nikmatnya Error



Sumber : pexels.com/@officialflavio

Ketika menghadapi error, mungkin reaksi kita kesal dan pusing. Tapi, kedepannya kita akan berterimakasih karena telah error. Error adalah salah satu nikmat programmer yang harus disyukuri. Dengan mengalami error, kita akan banyak belajar.

Awalnya, memang tidak semua orang senang dengan error. Sudah mencari sana sini tapi tetap error. Santai! Kamu tidak sendirian!

Akan error pada waktu nya.

Mulai dari junior hingga senior programmer pun pasti pernah mengalami yang namanya error. Yang membedakan adalah respon mereka kepada peristiwa itu.

Respon kita terhadap error atau bug yang berlebihan akan berakibatkan kita lelah dan tidak ingin melanjutkan coding. Jika error datang, nikmati, syukuri dan jangan terlalu membuat banyak drama di otak mu “Kok susah banget ya? Kok error mulu nih codingan gue!”

Jangan lupa untuk ber-istirahat sejenak sambil meminum secangkir kopi hangat.

Dokumentasi Adalah Sahabat Mu!



Sumber : [pexels.com/@wildlittlethingsphoto](https://www.pexels.com/@wildlittlethingsphoto)

Ketika Error melanda, terkadang kita ingin mengatasinya dengan cara cepat. Copy paste ke google dan error teratasi. Disatu sisi karena kita ingin cara cepat, kita melupakan logika dari program yang telah kita buat, yang penting errornya ngga ada dan terselesaikan.

Copy paste ke google, let's say kita dapat jawaban melalui stackoverflow. Itu sangat membantu. Tapi apa kamu tau versi berapa yg dia pakai? pada versi ini update apa saja yg terjadi?

Bisa jadi error dikarenakan kamu memakai versi baru dari salah satu framework yang kebetulan beberapa fitur di update. Sedangkan kamu menggunakan fitur lama yang sudah deprecated / tidak didukung oleh versi terbaru nya.

Semua ini balik lagi ke dokumentasi. Jangan hanya copy paste pesan error tanpa kamu tau letak spesifik errornya.

Fundamental



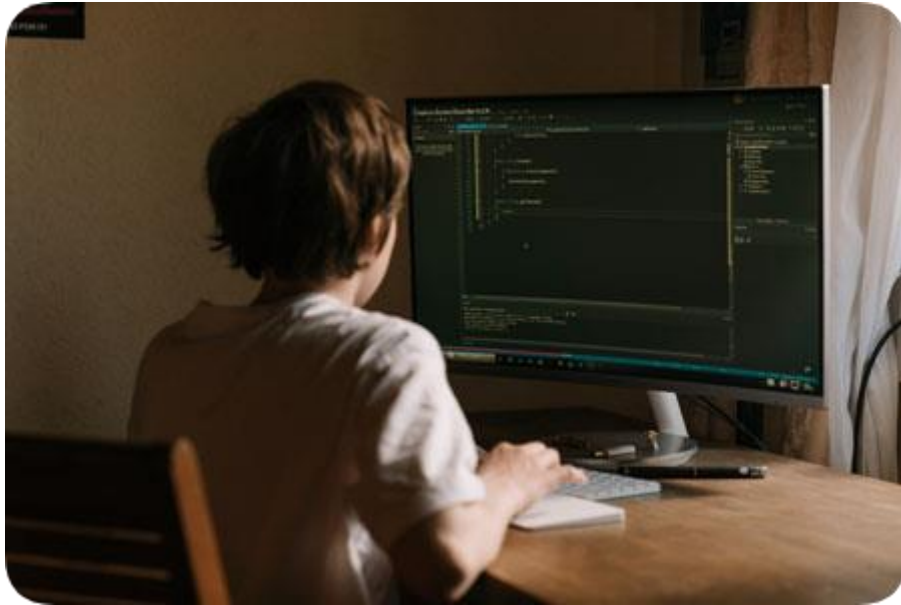
Sumber : [pexels.com/@kampus](https://www.pexels.com/@kampus)

Fundamental adalah prinsip dasar dari suatu ilmu. setiap ilmu memiliki fundamental nya masing". Begitupun dengan programming.

Tanpa fundamental, kita akan bingung. Contohnya waktu saya mempelajari codeigniter tanpa memahami PHP, dan konsep OOP (object oriented programming), jelas akan kebingungan.

Lebih baiknya kamu mempelajari fundamental dari bahasa pemrograman atau framework yg ingin kamu pelajari. Dan jangan lupa untuk membuat aplikasi sederhana dari apa yg sudah dipelajari. Hal ini bertujuan untuk mengasah apa yg sudah kamu pelajari.

DRY (Don't Repeat Yourself)



Sumber : pexels.com/@cottonbro

Bayangkan jika kamu memiliki 1 controller 1 file yang berisi header hingga footer. Tidak masalah jika untuk penggunaan landing page. Pertanyaannya " Bagaimana jika kamu memiliki 10 controller? atau 50? Bagaimana bila ada perubahan di navbar kita? seperti menambahkan sub menu..."

Jelas hal ini sangat tidak efisien. Kita tidak perlu mengulangi aktivitas (codingan) yang tidak perlu. Untuk penerapan dry, kita cukup memisahkan header, navbar dan footer kedalam satu folder. dan memanggilnya bila diperlukan.

Jika menggunakan Laravel, kamu dapat menggunakan template engine milik laravel, Blade.

API (Application Programming Interface)



Sumber : pexels.com/@thisisengineering

Contoh paling sederhana penggunaan API adalah aplikasi traveloka. Kok bisa ya kita beli tiket salah satu penerbangan di traveloka. Padahal traveloka dan perusahaan penerbangan itu berbeda sistem. Inilah fungsi dari APIs.

API (Application Programming Interface) adalah Interface yang dapat menggabungkan

aplikasi 1 dengan aplikasi 2. Simple nya, API itu komunikasi antar aplikasi.

Implementasi sederhana nya, kita ingin membuat sistem login menggunakan akun Google. Kita dapat menggunakan Google APIs nya. Bagi yang menggunakan Laravel dapat memilih Laravel Socialite.

Relasi Database

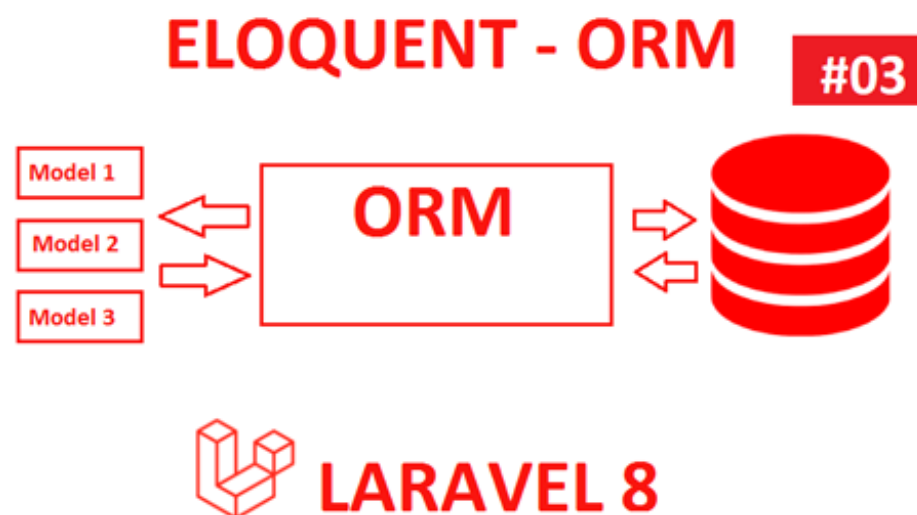


Sumber : openlogic.com/blog/what-sql-database

Orang Backend pasti sangat akrab dengan database. Ketika saya dulu baru belajar backend, hanya mengenal yang namanya CRUD (Create Read Update Delete).

Relasi database termasuk hal yang sangat penting karena hal ini dapat memudahkan komunikasi antar tabel.

ORM (Object Relational Mapping)



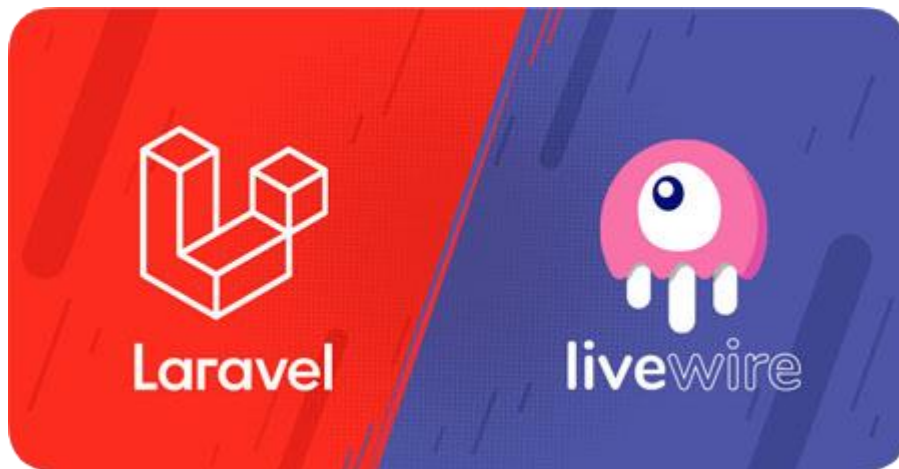
Sumber : norvicsoftware.com/eloquent-orm-en-laravel-8

Salah satu teknik favorit saya untuk keperluan backend. ORM merubah suatu table menjadi object sehingga kita dapat menggunakan konsep OOP.

Akan tidak efisien jika tidak menggunakan ORM. Contohnya kita menggunakan query builder,

jika terjadi perubahan nama column, kita akan merubah satu persatu secara manual.

Baru Kenal Livewire



Sumber : krishaweb.com

Pada saat itu saya membuat web app untuk keperluan pembagian rapot. Banyak yg bertanya "pak kok web e raport lemot ya, padahal aku cuma isi form aja lohh"

Awalnya saya menggunakan ajax. menurut saya ajax lebih cocok untuk membuat web menjadi interaktif. Secara default, Laravel mengirimkan data

ke server setelah itu ia melakukan refresh halaman atau reload data.

Ajax sedikit sulit hingga pada akhirnya saya menemukan Livewire. Livewire adalah framework fullstack untuk web yg dibuat menggunakan Laravel. Tujuan dibuatnya livewire agar menjadikan web kmu lebih interaktif.

Dengan Livewire, tidak perlu reload data. Sama dengan ajax, berjalan secara asinkronus.

STIGMA

“PANDANGAN MEREKA TENTANG PROGRAMMER”

Si Paling Bisa



Sumber : pexels.com/@gustavo-fring

Stigma ini paling sering dihadapi oleh anak IT. Anak IT seringnya dipandang si paling bisa memperbaiki apapun. Mulai dari perbaiki printer, wifi, komputer, dan lain-lain. Pernah mengalami? atau sedang mengalami?

Saya pernah disuruh memperbaiki mesin cuci saat bertamu di rumah teman. Komputer dengan mesin cuci beda jauh ya? Haha...

Padahal dalam dunia IT banyak sekali bidangnya. Web development, Mobile development, dan lain-lain. Di dalam web dev pun ada backend dan frontend. Antara backend dengan frontend pun berbeda framework.

Backend web ada Codeigniter, Laravel, Django, Symfony, CakePHP, Yii, dan masih banyak lainnya. Frontend ada Bootstrap, MaterializeCSS, Tailwindcss, Semantic UI, Vuejs, dan lain-lain.

Saya pernah mendapatkan project bersama teman kampus. Kebetulan dia baru memasuki dunia programming. Kami pernah miskomunikasi karena dia kira programmer itu semua fullstack.

Ketika dia membuat desain dari figma. Ia menyuruh saya untuk menjadikan design nya ke website. Saya bilang tidak bisa karena fokus saya di backend. Dan kalimat yang sempat dia ucap “Don’t mimpi ketinggian kalo ngga bisa eksekusi nya”. Mau dikata apa juga, yahhh namanya juga newbie.

Introvert



Sumber : pexels.com/@jeswin

Mungkin kamu pernah dikira introvert karena didepan komputer saja? Banyak yang bertanya "Lu introvert ye?". Jawabannya "IYA". Stigma ini benar untuk saya, namun mungkin bisa saja salah untuk kamu. Tidak semua programmer Introvert.

Menjadi Introvert atau Extrovert pun tidak masalah. Namun stigma programmer itu introvert dipandang orang lain sebagai orang yang hanya didepan komputer, irit berbicara. Biarkan mereka dengan pandangannya.

Tidak ada yang salah antara introvert maupun extrovert. Perbedaannya ada di cara mereka mendapatkan energi nya. Tipikal introvert akan terkuras energi nya jika bertemu dengan banyak orang. Namun bukan berarti introvert pendiam. Disisi lain, extrovert akan mendapatkan energi nya bila bertemu banyak orang.

“Lu introvert ye ru?” teman saya bertanya dengan wajah sinis. Seakan-akan introvert itu hal yang buruk. Pada saat itu saya berfikir bahwa menjadi introvert itu hal yang buruk.

Beberapa tahun kemudian setelah saya memiliki banyak teman, saya berfikir ulang “Apakah introvert itu buruk?” Sepertinya tidak buruk...

Saya pernah mencoba untuk menjadi extrovert namun tidak bisa. Alhasil saya lelah sendiri. Berbicara didepan banyak orang hal yang melelahkan, selayaknya berolahraga.

Jadilah diri sendiri...

Jomblo



Sumber : [pexels.com/@olly](https://www.pexels.com/@olly)

Bukan manusia bila tidak menyebalkan. terkadang beberapa teman saya bertanya "emang lu punya cewe bro? lu aje kerjaannya depan komputer". Yapssss.... Jawabannya "Saya tidak jomblo". Programmer dinilai kaku dan tidak cocok pacaran. Kerjaannya hanya didepan komputer dan ngoding. Namun tidak semua programmer jomblo ya.

Saya dan pasangan saya sudah bertahun-tahun menjalin hubungan alias pacaran. Tidak ada masalah bila saya ngoding, namun jika terlalu lama pasangan suka cerewet.

Memang pada awalnya saya sering dimarahi karena terlalu banyak ngoding. Tapi kesana-sana nya kami mengerti keadaan masing-masing.

Saya paling sering ditanya soal hubungan “Udah punya pasangan ru?”. Saya memiliki privasi untuk tidak menyebar informasi yang bersifat percintaan, keuangan, dan lokasi sekarang.

Kutu Buku



Sumber : pexels.com/@olly

Stigma kali ini mungkin relate untuk mu. Programmer, depan komputer, kacamata, buku, tutorial. Umumnya programmer berakhir menggunakan kacamata. Bukan hanya programmer saja, Jika kamu terlalu banyak didepan komputer maupun handphone pun saya rasa akan memakai kacamata.

Sebagai programmer kamu harus siap menggunakan kacamata (bila diperlukan). Banyak hal yang dilakukan didepan komputer.

Kurang Pergaulan



Sumber : pexels.com/@kindelmedia

Sering kali kaum programmer dipandang hanya menghabiskan banyak waktu didepan komputer. Tergantung pribadi masing-masing. Ada yang memang dia jarang bergaul. Ada juga yang memang senang bergaul. Namun disarankan untuk sering-sering nongkrong bersama teman atau bahasa keren nya 'hangout'.

Fakta nya di dunia IT bukan hanya coding saja, ada banyak kegiatan, seperti meeting, pertemuan dengan orang lain, dan lain-lain. Hangout bersama teman secara otomatis akan membuat kita melatih cara ber-komunikasi kita.

Kurangnya bergaul dapat membuat mu kesulitan dalam berkomunikasi entah dikantor maupun dirumah. Tidak ada salahnya untuk gaul bergaul bersama teman maupun orang lain.

ADVICE

“NASIHAT YANG MUNGKIN KAMU BUTUHKAN”

Meditasi



Sumber : pexels.com/@prasanthinturi

Seringkali kita menghadapi banyak error dalam sehari. Setiap programmer berbeda-beda dalam merespon error. Ada yg santai ada yg pusing.

Trus meditasi untuk apa? cara saya untuk bertahan di dunia Programming selama kurang lebih 5 tahun, ya dengan bermeditasi setiap hari nya.

Dilansir dari website alodokter "Meditasi adalah salah satu bentuk latihan untuk memusatkan dan menjernihkan pikiran, sehingga Anda bisa merasa lebih tenang, nyaman, dan produktif."

Semua programmer tidak ingin aplikasi buaatannya error. Error banyak, akhirnya stress. Meditasi adalah cara saya untuk menjernihkan pikiran, meredam stres, dan lebih produktif.

Disatu waktu, jika kamu mengalami error dan terus memaksakan diri untuk coding, saya pastikan aplikasi itu tidak efektif. Ada saja error yg merembet.

Berhenti sejenak, minum segelas teh atau kopi dan bermeditasilah selama 5 - 10 menit. Kamu dapat menggunakan teknik bermeditasi yg sering saya pakai. Teknik mengatur nafas dan menenangkan pikiran.

Keluar Sejenak



Sumber : pexels.com/@agisnu-andryan

Programmer tidak sepenuhnya di depan komputer mulu. Kita sebagai makhluk sosial sangat membutuhkan manusia lain.

Selama beberapa bulan, sekitar 2-3 bulan. Saya menarik diri sejenak dan memperbanyak belajar programming. Tindakan saya memang bagus.

Awal nya memang bagus hingga ketika saya menghadapi interview, saya mengerti bahwa tidak

baik jika ngoding mulu. Dampaknya saya tidak bisa menangkap secara baik apa yg mereka obrolkan.

Keluarlah sejenak, hirup udara segar. Berbicaralah bersama manusia, bukan komputer atau laptop. Sebaik nya kita berkomunikasi dengan komputer atau laptop dengan takaran secukupnya. Coding ketika dibutuhkan.

Terlalu banyak ngoding pun tidak baik untuk perkembangan hidup kita. Banyak hal diluar sana yang menarik untuk diexplore. Pegunungan, sungai, puncak, taman rekreasi.

Semua baik adanya bila sesuai dengan takaran cukup dan tidak berlebihan.

Jangan FOMO



Sumber : pexels.com/@katlovessteve

Fear of Missing Out atau FOMO adalah seseorang yang takut akan tertinggal segala hal. Pada awal tahun memasuki dunia IT, saya seringkali dihadapkan oleh fomo namun saya tidak tahu bahwa itu fomo.

Semua hal ingin saya pelajari dan perdalam saking takutnya tertinggal zaman. Bootstrap, JQuery, Codeigniter, Laravel, MaterializeCSS semua saya pelajari dalam 1 waktu.

Setelah 1 tahun saya sadar bahwa saya sudah fomo melihat teman saya sudah bisa framework A, B, C. Alhasil Saya hanya mengerti Bootstrap. Fomo dapat menggerakkan kamu menuju kerakusan. Bukan hanya kerakusan tapi kerugian waktu.

Pada dasarnya tanpa kita fomo kita sudah ketinggalan zaman, teknologi berkembang secara pesat. Fomo sangat bisa memicu munculnya rasa tamak di banyak hal.

Ketika kamu belajar susah payah karena ingin mengikuti zaman atau supaya tidak ketinggalan teman kamu namun sedikit yang kamu dapat, bisa jadi kamu Fomo.

Sadar akan sedang mengalami fomo, saya mencoba untuk mengatur ulang jadwal belajar programming saya secara teratur. Jangan menargetkan target belajar diluar batas kemampuan kamu.

Kenali Batasan Kemampuan Mu.

Puasa Sosial Media



Sumber : [pexels.com/@tracy-le-blanc](https://www.pexels.com/@tracy-le-blanc)

Sepi job hal yang biasa untuk programmer. Semua programmer freelance pernah mengalami itu semua. Ketika awal freelance, hal yang sering saya lakukan ketika sepi job adalah bermain instagram.

Semua berjalan baik adanya hingga saya cukup risih dengan diri saya alias insecure. Kok bisa? bisa. Hal tersebut disebabkan oleh banyaknya konten yang pamer.

Punya mobil di posting, punya pasangan di posting, semua hal di posting. Pernah melihat orang seperti ini? atau kamu pernah insecure ketika scrolling instagram atau sosial media lainnya? Kok bisa ya dia hidupnya enak, lah aku?

Jika kamu risih atau insecure akan konten konten di sosial media, silahkan **puasa sosial media**. Di awal latihan, coba batasi penggunaan sosial media secara perlahan. Jika awalnya 5jam coba untuk dikurangi setengah jam.

Gunakan aplikasi untuk keperluan screen time supaya kamu dapat melihat perkembangan latihan kamu. Hindari untuk langsung ke tahap selanjutnya : ***uninstall sosial media***.

Ketika kamu merasa sudah sangat cukup latihan mengurangi jam sosial media kamu, silahkan masuk ketahap selanjutnya, yaitu uninstall sosial media.

Terusss? coba untuk install hanya untuk 1 minggu sekali, semisal kamu install setiap minggu. senin nya kamu uninstall kembali. Teruskan selama beberapa bulan, coba untuk perkecil menjadi 2 minggu sekali, hingga 1 bulan sekali.

Kamu akan merasakan lebih cinta dan damai terhadap diri sendiri. Menjalani aktivitas menjadi lebih enak dan produktif.

THANK'S TO

Terimakasih kepada pihak yang sudah membantu dalam pembuatan e-book ini,

dosen saya,

senior-senior saya, dan

teman-teman saya.