

>

—

# Hadiah Untuk Programmer

Hilman Ramadhan

Founder sekolahkoding

# Daftar Isi

## #Halo semuanya!

- Kenapa menulis buku ini?
- Berkenalan dengan koding
- Surat takdir
- Belajar di luar kuliah
- Kenapa kamu mau jadi Programmer ?

## #Membuka pintu ilmu

- Skill bukan mie instan
- Mulai dari akhir
- Belajar dengan materi bertahap
- Mencuri seperti seniman
- Newbie ngga perlu minta maaf
- Bereksperimen seperti ilmuwan
- Surat cinta untuk masa depan

## #Berjalan dengan masalah

- Error itu proses
- Umur yang cocok untuk ngoding
- Matematika saya jelek
- Programming ngga sendiri
- Mengikuti teknologi terbaru
- Satu laptop seumur hidup
- Bisikan ngga pantas akan datang
- Susah itu pertanda bagus
- Localhost memang aman
- Kamu akan lelah

## #Versi terbaik kamu

- Belajar meneliti
- Belajar bahasa Inggris
- Boleh deh tapi free kan
- Berpikir Garis besar
- Refactoring dan Eliminasi
- Don't repeat yourself
- Skill menyelesaikan masalah
- Bicara dan bantu orang lain

## #Rezeki di dunia koding

- Butuh Sarjana dan setifikat
- Siapkan Portofoliomu
- Bekerja tetap atau sebagai freelancer
- Programmer bikin bisnis sendiri?

## #Sekarang giliran kamu

- Tool yang paling bagus
- Kode yang sempurna
- Sapi warna ungu
- Apa hadiahmu?

**#Halo  
Semuanya!**

# Kenapa Menulis Buku Ini?

Kenalkan saya Hilman Ramadhan, saat ini sharing seputar programming di [sekolahkoding](#) dan ngobrol seputar kehidupan lewat podcast + tulisan di [tehataukopi](#).

Saya menulis buku ini sebagai hadiah untuk kamu. Seperti hadiah pada umumnya, saya ngga tahu kamu akan suka atau ngga. Ini hadiah, ngga perlu kamu bayar alias gratis. Tapi gratis bukan berarti ngga berharga.

Saya berharap isinya ngga akan terbatas dengan waktu, ngga relevan dengan teknologi yang sedang “hits”. Buku ini bukan tutorial, tapi apa yang bisa saya bagikan dari dunia koding, bagaimana kamu bisa terjun, bertahan di dunia ini dan mengubah dunia jadi sedikit lebih baik!

Saya ngga tahu kapan saya meninggal, semoga tulisan ini menjadi salah satu warisan terbaik selama dan setelah saya hidup!

# Berkenalan Dengan Koding

Sebelum masuk kuliah di Jerman, saat itu saya perlu melewati “Studienkolleg”, kelas persiapan untuk pelajar dari luar Jerman yang ingin kuliah di sana selama satu tahun, kembali mengulang beberapa pelajaran di SMA.

Karena mengambil jurusan IPA, harus bertemu lagi dengan kimia. Pelajaran yang entah kenapa seperti ada segel terlarang yang saya ngga bisa masuk ke dalamnya. Berkali-kali saat pelajaran saya harus “izin ke toilet”, hanya untuk duduk di tangga menunggu pelajaran selesai.

Salah satu triknya, karena di kelas kimia selalu ada latihan soal yang harus dikerjakan di papan tulis, saya akan angkat tangan di awal, karena di awal soalnya lebih gampang. Kalau sudah naik hari itu, tidak perlu lagi khawatir akan ditunjuk untuk kedua kalinya.

Di semester dua “Studienkolleg” kali ini saya bertemu dengan pelajaran “Informatik” (Informatika), disebutlah “Pascal”, saya kira nama orang pintar jaman dulu sekelas Aristoteles ternyata Pascal adalah salah satu bahasa program.

Salah satu tugas yang harus dikerjakan waktu itu mengubah hitungan waktu jam menjadi detik. Bereksperimen di terminal / Command-prompt, membuat program sederhana ini dengan Pascal, benar-benar membuat saya jatuh cinta dengan programming.

Jujur ngga semuanya menarik, ada waktu harus mengerjakan tugas rangkaian TRUE dan FALSE, menghafal 1 bit berapa byte dan teori lainnya, yang saat itu saya ngga sadar akan sangat membantu beberapa tahun kemudian!

Saat masih kecil saya tahu jawaban ketika ditanya “mau jadi apa?” pilihannya antara astronot atau pemain bola. Tapi tidak saat itu, saya ngga tahu harus jadi apa, harus kuliah apa, yang saya tahu, senior asal Indonesia disana banyak yang mengambil Teknis Mesin. Beberapa bulan awal hanya jurusan ini yang ada di pikiran saya.

Karena dibolehkan memilih dua jurusan saat melamar kuliah, jurusan pertama yang saya pilih adalah teknis mesin, karena sudah ada senior yang bisa membantu nanti. Informatika ada di urutan kedua terlepas dari hati saya jauh lebih cenderung memilih jurusan ini.

# Surat Takdir

Surat balasan dari universitas datang! tertulis “TIDAK LULUS”, kuliah di jurusan teknik mesin tinggal bayang-bayang ilusi. Jelas ada perasaan kecewa dan sedih, yang kalau sekarang mundur ke belakang, entah apa yang terjadi kalau saat itu diterima di teknik mesin. Tersisa informatika, petualangan menarik di dunia kuliah akan dimulai !

Saat itu yang dipelajari sebagian besar adalah bahasa program Java, salah satu tugasnya saya perlu mengimplementasikan rumus yang diberikan ke sebuah aplikasi mengedit gambar. Seperti rumus mendeteksi wajah atau filter warna di instagram, rumusnya diajarkan di kelas dan perlu menambahkan fungsinya di aplikasi editor gambar. Sejauh ini semuanya menarik.

Semester satu, dua dan tiga terasa indah ~, kuliah seperti taman bunga, semuanya wangi dan membuat betah. Sampai di semester ke empat saya mulai merasa apa yang dipelajari ngga bisa saya praktekan, saya ngga bisa bikin sesuatu.

Saya mulai ngga serius kuliah, memilih datang ke kuliah hanya pelajaran tertentu. Salah satu pelajaran yang tidak menarik untuk saya saat itu adalah “computer graphic” di



mana kita harus membuat gambar atau animasi melalui bahasa program.

Ada aturan di sebagian besar kampus di Jerman, 3 kali tidak lulus pelajaran yang sama, harus di drop-out. Seperti yang teman-teman bisa tebak, pelajaran “computer graphic” jadi pintu masuk saya untuk keluar dari dunia perkuliahan.

Setelah ujian ke 3, dengan perasaan tidak tenang saya membuka emailnya, tertulis “TIDAK LULUS!” yap, saya harus keluar dari kuliah, berbeda seperti surat pertama, dimana ada “cadangan” memilih jurusan kuliah, kali ini TIDAK ADA PILIHAN LAIN.

# Belajar di Luar Kuliah

Masa-masa suram versi “saya saat itu”, dikeluarkan dari kuliah, semakin ngga jelas mau jadi apa nanti. Satu hal yang saya tahu, saya suka dengan programming. Saya suka bikin sesuatu. Masa-masa “lulus kuliah” saya gunakan belajar programming khususnya dunia website. Mulai bergriya di internet, mencari sumber belajar. Youtube, ebook, artikel semuanya terlihat menarik.

Saya mulai membuat website, ada dua yang saya ingat, (1) website untuk mengatur formasi pemain juventus(salah satu klub bola asal Italia), dimana user bisa drag/drop nama-nama pemain bola, mengatur formasinya sendiri, mengambil screenshot dan menguploadnya (2) website kumpulan media islami (ada youtube, audio, twitter dll). Bisa dipastikan tidak ada yang mengunjungi apalagi berkomentar di websitenya.

Keluarga mulai mendengar kabar saya membuat website, dari sana saya dapat pekerjaan dua website usaha keluarga. Saat itu saya ngga memikirkan harga, bahkan tidak ada diskusi tentang ini sebelum dimulai, saya hanya senang bisa mempraktekkan pekerjaannya.

Kedua website ini saya jadikan portofolio untuk melamar kerja di Indonesia dan di Berlin, Jerman. Sampai di beberapa lamaran saya menulis “rela tidak dibayar yang penting pengalaman”. Mungkin ada 20an perusahaan di Berlin, semua balasan emailnya sama “mohon maaf karena banyaknya pelamar lain yang lebih memenuhi kualifikasi ..... anda TIDAK LULUS” saya sudah familiar dengan kalimat ini.

Sampai pada saat sudah menyerah, saya ngga lagi mengirim lamaran, ada satu perusahaan yang mengajak wawancara dan tes pekerjaan. Kali ini lulus! saya bertugas sebagai frontend web developer yang fokusnya membawa website perusahaan menjadi responsive (baik dilihat di smartpone). Dan berita baiknya... digaji!! Tidak disangka karena sebelumnya sampai menulis tidak perlu dibayar.

Tempat itu jadi satu-satunya tempat saya pernah bekerja dan pernah punya bos sampai saat ini. Selamat satu tahun saya belajar dari pengalaman dan dari teman-teman, yang berasal dari Rusia, Jerman dan negara lain.

Di waktu yang sama, meskipun saya tidak lulus kuliah, beberapa teman yang kuliahnya ada pelajaran informatika, masih percaya untuk belajar di masa ujian dengan saya. Setiap ingin memberikan rekomendasi sumber belajar dalam bahasa Indonesia, saya ngga menemukan yang bagus dari internet.

Dari sinilah awalnya sekolah coding lahir, yang menjadi kesibukan saya sampai hari ini. Saya yakin ilmu ini sangat penting. Kita (Indonesia) akan kembali ketinggalan kalau ngga ada yang serius terjun ke bagian edukasi di bidang programming.

# Kenapa Kamu Jadi Programmer?

Mencari tutorial dan bagaimana cara ngerjain sesuatu di jamanNow sangat gampang, segel internet terbuka untuk semua orang. Sayangnya banyak yang lupa alasan KENAPA mengerjakan sesuatu.

Bagaimana dengan kamu? kenapa mau jadi programmer? karena terlanjur masuk jurusan informatika? karena startup digital sedang populer? Gimana kalau kamu ngga di jurusan informatika atau perusahaan-perusahaan digital ngga lagi populer? masih mau jadi programmer?

Saya ngga tahu ada alasan yang benar atau salah, yang pasti kalau alasan menjadi programmer hanya untuk uang atau populer, ada banyak pekerjaan lain yang bisa dijadikan alternatif. Tapi kalau kamu percaya dengan membuat website atau aplikasi ada banyak manfaat yang kamu kontribusikan ke dunia ini, SELAMAT DATANG!

Bukan berarti semuanya akan jadi mudah, bahkan jauh dari itu, akan banyak masalah dan tantangan. Berkali-kali kamu akan merasa “ngga pantas jadi programmer”, “ngga cocok”, “kurang pintar” dan segudang alasan lainnya.

Terlepas dari semua hal ngga mungkin dan keterbatasan yang akan kamu alami, akan selalu ada jalan dan akan banyak teman yang siap menemani (loh, bukannya programmer hidupnya sendiri? ngga dong! :D, akan dibahas di bagian lain) .

Alasan saya saat memulai [sekolahkoding.com](http://sekolahkoding.com) sendiri karena kurangnya tutorial dalam Bahasa Indonesia saat itu (2015). Orang-orang bingung harus mulai dari mana dan harus ke mana. Dengan harapan akan banyak lahir karya-karya untuk menyelesaikan berbagai masalah di Indonesia dan dunia melalui aplikasi yang dibuat.

Sampai saat buku ini saya tulis, saya masih ngoding, masih terus berhadapan dengan text-editor, semuanya masih menarik dan seru seperti saat saya pertama kali jatuh-cinta dengan programming. Bagaimana dengan kamu, apa alasan kamu mau jadi programmer ?

# #Membuka Pintu Ilmu

# Skill Bukan Mie Instan

Mau dipuji? tinggal ke instagram upload foto makanan kamu. Mau dapat likes? tinggal ke Facebook kutip kata-kata orang hebat. Pernah dengar ada yang beli kunci jawaban Ujian Nasional? atau bayar sekian rupiah untuk lolos menjadi Pegawai Negeri Sipil? Hampir semuanya serba instan di negeri tercinta kita ini.

Ketika hasil bisa dibeli dengan instan, “Skill” jadi hal yang sangat langka, hanya beberapa orang yang mau Lewatin proses untuk mendapatkan ini.

Dan tentunya orang-orang langka yang banyak dicari.

Kalau kamu hari ini punya banyak uang, besok uangnya hilang, semuanya hilang. Kalau kamu punya skill, uang kamu hilang, kamu bisa kembali nyari uang dengan skill yang kamu punya.

Hilangkan jauh-jauh pikiran “jadi ahli programming dalam 1 hari”. Judul kursus atau judul buku mungkin banyak yang seperti ini, tapi kenyatannya ngga. Satu hari mungkin cukup untuk perkenalan, bukan untuk menjadi ahli.



Harus mau merasakan lelah, mau berpikir lebih serius dari biasanya. Akan ada karya-karya atau kode yang kamu mungkin malu ngelihatnya di masa depan. Tapi itu bagian dari perjalanan.

# Mulai Dari Akhir

Salah satu saran terbaik yang pernah saya dapat adalah “mulai dari akhir”. Bayangkan akhir seperti apa yang kita mau, mulai dari sana.

Kalau kamu mau mulai belajar ngoding, tahu persis apa yang mau kamu bikin. Belajar dengan bikin sesuatu akan menambah fokus dan bahan bakar energi selama belajar.

Selain dengan mengingat tujuan akhirnya, teknik ini juga bisa dipakai di skala kecil, saat kita belajar. Misalnya hari ini target saya “membuat menu navigasi yang ada di website”. Tuliskan di kertas dan setelah merasa selesai belajar, coba bikin menu navigasi seperti yang kamu mau. Tujuannya jelas, proses yang harus dipelajari pun jadi jelas.

Gimana kalau kamu mau kerja? bukan mau bikin sesuatu? ngga masalah, buka website tempat di mana kamu mau kerja, buka bagian karir atau lowongan kerjanya, lihat skill apa aja yang harus kamu punya. Jadikan ini daftar “mata kuliah mandiri”. Mulai dari akhir.

# Belajar Dengan Materi Bertahap

Belajar dari satu materi ke materi lain, dengan membaca atau menonton, tidak ada masalah. Terutama kalau kamu sama sekali nggak ada pengalaman, nggak tahu mau mulai dari mana, kamu bisa mengikuti materi yang udah ada di internet.

Dengan tujuan sebagai tempat awal, bukan menjadikan peta yang semuanya sudah jelas, tapi jadi kompas penunjuk arah harus berjalan ke mana. Mengetahu dasarnya, tujuannya bukan untuk dihafal. Banyak yang mengeluh nggak bisa menghafal materinya, padahal hafal sama sekali bukan tujuan. Tujuan kita membaca atau materi biar kita familiar dengan syntax atau bahasanya.

Sama sekali tidak ada masalah melihat dokumentasi, justru dokumentasi dibuat untuk dibaca. Jadi jangan pusing kalau kamu nggak hafal metode tertentu. Semuanya akan jadi kebiasaan secara otomatis.

Jangan lupa, gerakkan jarinya. Jangan hanya klik sana klik sini, download ebook atau video yang akhirnya cuman menumpuk di komputer. Praktekkan yang dipelajari, ikuti, coba tulis ulang apa kode yang kamu pelajari.

# Mencuri Seperti Seniman

Buku “Steal like an artist” yang ditulis Austin Kleon ngajarin gimana para seniman mencari beberapa karya yang bisa kamu contoh untuk jadi bahan belajar atau referensi berkarya. Di programming pun ini berlaku, cari kode yang kamu suka di internet, misalnya di GitHub, ada banyak open-source yang tersedia. Buka file-filenya lihat kode yang orang lain tulis.

Kalau ada library atau framework yang kamu suka, cari tahu bagaimana caranya dibikin. Ikuti apa yang ada disana, bikin kasus seperti bagaimana kalau saya mau menambahkan fungsi baru, file mana yang harus saya tambahkan, di mana kodenya harus saya tulis.

Hampir semua kode yang mau kita buat sudah pernah dibuat sebelumnya. Coba cari contohnya, jangan hanya copy-paste lalu mengganti kontennya, tapi pelajari, apa yang bisa diperbaiki, pelajaran apa yang bisa kamu ambil dari kode orang lain.

# Newbie Ngga Perlu Minta Maaf

Kalimat yang biasa keluar di forum-forum programming saat bertanya adalah “maaf saya masih newbie”. Semua orang pernah newbie alias baru memulai, ngga perlu minta maaf, ini bukan kesalahan.

Kamu boleh bertanya, tapi jangan lupa berusaha sebelumnya. Yang menjadi masalah ketika kita serba mau jawaban instan. Setiap ada masalah jangan langsung bertanya, tapi coba selesaikan sendiri sebelumnya.

Banyak yang bertanya dengan menganggap orang lain seperti dukun, “mas saya mau buat toko online, tapi ko’ error ya? salahnya dimana?” orang lain jelas ngga tahu salahnya dimana, karena ada 1001 kemungkinan kesalahan yang kamu lakukan.

Bertanya dengan sopan, di dunia nyata atau dunia maya. Jelaskan apa yang sedang kamu kerjakan, usaha apa yang udah kamu lakukan dan masalahnya dimana.

# Bereksperimen Seperti Ilmuwan

Ilmuwan tahu, ketika ada percobaan yang ngga berhasil, dia bukan gagal, tapi menemukan cara yang salah.

Jangan takut bereksperimen dengan koding. Kamu ngga akan rugi, hal yang paling jelek yang bisa terjadi saat proses belajar, adalah error. Dimana error ini bisa diperbaiki dan menambah pengetahuan kamu.

Sebelum kamu bertanya di forum-forum “bisa ngga ya kalau..” langsung dicoba! cari tahu jawabannya bisa atau ngga. Setelah itu diskusikan dengan teman-teman lain di forum, kenapa bisa atau kenapa ngga bisa.

Ilmuwan juga ngga bosan dengan berkesperimen. Di dunia koding, ngga perlu nunggu berbulan-bulan, setiap hari pun akan selalu ada tool baru, ada library, plugin atau framework baru. Jangan takut untuk bermain dengan semua alat ini.

Lihat kodenya, seperti apa tool ini dibuat. Bandingkan apa yang bagus dari tool satu dengan tool yang lain. Yang lebih menarik lagi, bisa ngga kamu bikin tool seperti ini ?

# Surat Cinta Untuk Masa Depan

Damian Conway pernah bilang “Dokumentasi adalah surat cinta kamu untuk masa depan”. Yang saya bicarakan disini bukan dokumentasi dari orang lain. Tapi dokumentasi selama kamu belajar.

Seperti menulis jurnal, bisa di komputer atau tulis tangan, apa yang mau kamu pelajari, pecah jadi bagian-bagian kecil. Tulis masing-masing catatan untuk bagian kecil ini, apa yang kamu temukan setelah belajar.

Termasuk ketika ada error, ini kesempatan emas, untuk ngga mengulang kesalahan yang sama di masa depan. Tulis errornya, tulis bagaimana solusi menyelesaikan error ini. Sewaktu-waktu saat kamu ketemu masalah yang sama tinggal membuka surat cinta ini.

# #Berjalan Dengan Masalah



# Error Itu Proses

Bug alias error akan menjadi teman setia kita selama ngoding, bukan hanya saat belajar tapi saat bekerja pun, bug ngga akan meninggalkan kita sendiri. Bisa jadi waktu yang kita pakai berjam-jam sampai stress hanya karena lupa menggunakan tanda koma saat memisahkan sesuatu. Bisa jadi kesalahan fatal seperti logika program, penggunaan if-else yang terbalik dan error-error lainnya.

Tenang! Baca error dengan tenang, jangan panik. Sebagian besar bahasa program menyediakan laporan error yang jelas. Apa kemungkinan kesalahan yang dibuat dan pada baris berapa kesalahan ini terjadi.

Belajar membaca error, tanpa terburu-buru. Dokter yang panik ngga akan bisa memberikan obat yang benar ke pasiennya. Asal suntik bagian yang salah hanya akan menambah derita pasiennya. Asal mengubah kode tanpa melihat betul masalahnya, hanya akan membawa error-error lainnya.

Belajar “debug” atau mencari error dengan teliti. Contoh ketika kamu punya 3 fungsi yang saling berhubungan. Belum tentu errornya ada di fungsi yang ketiga, bisa jadi

kesalahannya ada di fungsi pertama. Untuk itu perlu mencari akar dari kode yang kita tulis. Periksa dengan teliti kemungkinannya. Hentikan program di fungsi satu dan lihat hasilnya, kalau tidak ada masalah, coba ke fungsi 2, coba ke fungsi 3 dan seterusnya.

Jadikan error seperti rival. Setiap dia muncul, bilang dalam hati, ada satu ilmu baru lagi yang akan saya dapat, ada satu tantangan lagi yang siap saya lewati.

# Umur yang Cocok Untuk Ngoding

Pertanyaan ini ngga kalah populer, Umur yang cocok untuk ngoding adalah umur 3-6 bulan pada saat masih bayi, mumpung masih segar. Tenang, hanya bercanda!

Tidak ada umur yang kaku untuk memulai, ketika kamu udah sadar kenapa mau ngoding dan mau berkarya dengan ini, jangan lagi pedulikan hal-hal tidak penting seperti umur. Ngga akan ada yang peduli berapa umur kamu saat mau belajar ngoding atau membuat karya. Tentu berbeda kalau pertanyannya umur berapa batas kerja di perusahaan, karena kebijakan ini tergantung dari perusahaannya.

Tapi untuk belajar dan berkarya, mulai sekarang! jangan bilang “waduh saya masih terlalu muda, nanti aja” atau “hmm udah tua, kayannya ngga cocok”. Ini semua cuman salah satu dari berbagai macam bisikan yang akan datang.

Komputer atau laptop kamu ngga akan marah dan bilang “Hey bocah ingusan jangan coba-coba” atau “Pak Tua, pergi dari sini, kamu ngga akan sanggup” ini semua hanya ilusi ~.

# Matematika Saya Jelek

Sama saya juga! Meskipun selama kuliah saya selalu percaya diri ketika ujian matematika, sayangnya hasilnya berkata lain, nilainya selalu jelek entah kenapa, mungkin ada segel terlarang juga yang saya ngga boleh masuk seperti pelajaran kimia.

Ketika kamu membuat program, tentu akan bertemu logika-logika yang perlu diimplementasikan pada kasus tertentu, tergantung apa yang kamu buat, akan ada rumus-rumus matematika yang perlu kita terjemahkan dengan bahasa program.

Senior saya pernah menceritakan, salah satu kutipan yang pernah dia baca yang sangat menginspirasi hidupnya dan juga sekarang hidup saya “Yang mau akan mencari jalan, yang tidak akan mencari alasan”. Kita bisa memberikan lampu hijau ke diri kita, boleh menyerah, atau kalau kita benar-benar mau, cari jalannya.

Tenang, semuanya bisa dipelajari, ngga perlu jadi expert di bidang ini, cukup mau belajar dan mau cari tahu sesuai kasus yang kamu temukan sekarang. Rumus-rumus sudah ada di internet, ketika ada logika yang sedikit sulit, jangan ragu untuk tanyakan ke yang lebih ahli. Bukan hanya ahli di bidang programming, tapi ahli di bidang masalah yang kamu hadapi.

Misalnya kamu ada pekerjaan membuat aplikasi untuk perhitungan di dunia arsitek. Jangan hanya ngobrol dengan komputer, tapi ngobrol juga dengan arsitek-arsiteknya. Kenali pandangan mereka seperti apa, ketika ada masalah, tanyakan apakah teorinya di atas kertas sudah benar atau belum, tanpa mereka harus tahu bagaimana kamu implementasikannya di program.

# Programming Ngga Sendiri

Mitos yang sering beredar di masyarakat, programmer hidupnya sendiri, di gurun, ngga ada apa-apa, ngga ada siapa-siapa. Tenang aja, ini juga gosip. Saya kenal banyak programmer yang sudah berkeluarga dengan anak-anak yang ceria, jadi jelas ini hanya mitos.

Seperti yang disampaikan di bagian sebelumnya, programmer perlu mengerti alasan dan apa yang sedang dibuat. Perlu komunikasi ke orang-orang yang sesuai dengan tema yang dibuat. Jangan berharap akan ada pengguna website atau aplikasi kita secara otomatis, kalau kita ngga kenal masalah yang kita berusaha buat solusinya ini.

Kamu punya orang-orang yang sudah lebih dulu melewati masalahnya yang siap membantu, kamu punya orang-orang yang siap berjuang bersama-sama dan ada juga orang-orang yang akan datang yang baru mau belajar butuh bantuan kamu. Dulu, sekarang dan masa depan akan penuh dengan orang lain.

Komunitas programmer pun menjadi salah satu komunitas yang sangat aktif . Selain forum-forum yang ada di dunia maya, ada banyak konferensi di skala besar perusahaan, meetup-meetup kecil di kota-kota atau study-club di beberapa kampus. Tentunya anggotanya manusia, bukan butiran pasir.

“Tapi di kota saya ngga ada!” ingat jangan cari alasan tapi cari jalan. 1. Kamu bisa keluar kota, bergabung dengan komunitas di kota terdekat dan no. 2 kamu bisa memulai, membuat komunitasnya di kota kamu. Mulai dari 2-3 orang sampai ruangan kamu nanti ngga sanggup menampung semua anggotanya.

Pada saat teman-teman bekerja di suatu perusahaan pun, akan ada senior, orang yang lebih dulu mengerjakan programnya, kalau ngga mau tersesat dan merusak program perusahaan di hari pertama, jelas perlu komunikasi dengan sang senior. Seperti apa alur programnya, masalah struktur, problem sejauh ini dan hal-hal yang perlu diperhatikan lainnya.

Belum lagi dengan designer, konflik akan banyak terjadi kalau komunikasi ngga bagus. Masalah kecil seperti ukuran icon yang ingin diperbesar, warna detail dalam hexacolor, dan segudang pekerjaan lainnya, akan mengubah suasana pekerjaan jadi ngga nyaman kalau kita ngga berkomunikasi. Masih ada lagi hubungan denga bos, klien atau pengguna.

Mencari partner di level yang sama juga perlu, bukan hanya yang levelnya di atas atau levelnya di bawah, tapi punya sosok rival, akan menambah semangat untuk terus improve skill yang kamu punya. Sama-sama terus berkembang!



# Mengikuti Teknologi Terbaru

Akan selalu datang teknologi baru, sebagai pelajar yang baik saya yakin kita mau nyobain semuanya, sayangnya waktu untuk bekerja dan belajar sering tidak bersahabat. Ini yang sering membuat dilema.

Pertanyaan yang muncul “gimana saya nyobain semuanya? waktunya ngga akan cukup!” betul sekali! waktunya ngga akan cukup dan berita baiknya, kamu ngga perlu nyobain semuanya. Jangan latah dengan yang terbaru, tapi yang bikin kamu produktif.

Bruce Lee pernah bilang (saya belum pernah dengar Bruce Lee langsung, tapi ini yang saya temukan di internet), “saya lebih takut dengan lawan yang punya 1 jurus tapi latihan 1000 kali, dibanding yang punya 1000 jurus latihannya hanya satu kali”.

Kita ngga bisa hanya latah, pindah dari satu teknologi ke teknologi lain, kecuali teman-teman penulis blog, vlogger atau pengamat yang memang pekerjaannya melakukan ini. Karena kita selalu punya pekerjaan yang perlu diselesaikan.

Belajar teknologi baru memang asyik, tapi ingat ada prioritas yang perlu diselesaikan.

Bukan berarti kamu ngga boleh belajar teknologi baru, justru ini menjadi kewajiban. Kalau kamu punya waktu luang, membandingkan tool yang satu dan yang lain akan bisa membuka pikiran.

Pertanyaan yang ngga boleh kelewatan “bagaimana yang kamu pelajari diimplementasikan di dunia nyata?”. Kalau kamu ingin mengerjakan sesuatu yang produktif atau bekerja dengan orang lain, perlu komitmen untuk menggunakan satu teknologi / bahasa / framework yang didalami, agar fungsi dari teknologi yang kita pilih bisa dimaksimalkan.

Kalau kita hanya melihat laut dari atas, kita ngga akan bisa menghargai keindahan di bawahnya, yang tentunya jauh lebih menarik. Orang-orang berdebat, yang ini lebih baik dari itu, padahal belum mencoba mendalaminya. Saya yakin ngga ada satu tool yang sempurna untuk semua pekerjaan dan untuk semua orang.

Kamu akan mencari referensi di internet, teknologi mana yang terbaik dan menemukan semua orang punya pendapat masing-masing. Disini penting untuk melihat sumber aslinya, baca dokumentasi atau website resmi dari teknologi ini dan baca tujuan tool ini dibuat. Bandingkan dengan tujuan kamu, apakah ini cocok atau ngga.

Beberapa tips dalam mencari tool yang cocok untuk kebutuhan:

1. Lihat dokumentasi resminya, tujuan tool ini untuk apa
2. Banyaknya komunitas, karena sewaktu-waktu kita akan menemukan masalah dan butuh bantuan dari komunitas ini, baik offline maupun online.
3. Lihat contoh kasus (aplikasi atau website) yang sudah dibuat dengan tool ini.
4. Coba bereksperimen sendiri!

# Satu Laptop Seumur Hidup

Gimana kalau kamu cuma boleh punya satu laptop seumur hidup? saya yakin laptopnya akan dirawat dengan baik. Dibersihkan, hati-hati dalam memasukkan program di dalam agar tidak terkena virus dan menjaga dengan aman di luar agar tidak rusak apalagi hilang. Mungkin kita akan punya lebih dari satu laptop selama hidup, tapi tidak dengan badan kita. Selama hidup kita hanya punya 1 badan ini.

Sayangnya kita ngga memperlakukan badan kita seperti barang yang mahal, seperti barang yang kita sayangi. Kita ngga peduli makanan yang masuk sehat atau ngga, belum lagi rokok, barang yang jelas-jelas di bungkusnya sendiri tertulis banyak potensi penyakit berbahaya.

Yang jatuh cinta dengan minuman hangat seperti kopi atau yang dingin seperti minuman bersoda akan lupa dengan pentingnya minum air putih yang cukup. Konsumsi minuman tadi secara berlebihan, efek pertamanya akan menjadi kebiasaan, kita jadi ngga bisa “ngoding” kalau ngga minum minuman tadi. Belum lagi berbicara masalah kesehatan organ tubuh kita.

“Mumpung masih muda” katanya, justru karena masih muda, ini waktunya kita untuk menjaga kesehatan dengan baik. Organ-organ yang kita dapatkan di dalam tubuh memang gratis, tapi apakah gratis berarti murah atau ngga berharga?

Kebiasaan lainnya yang mengganggu kesehatan adalah terlalu banyak duduk, karena merasa pekerjaannya penting. Betul pekerjaan penting, tapi menjaga kesehatan juga penting. Jadwalkan rutin mengangkat pantat kamu dari kursi saat bekerja.

Salah satu teknik yang populer adalah teknik pomodoro. Dimana kita menyiapkan waktu 25 menit untuk fokus bekerja dan 5 menit beristirahat. Teknik ini diulang-ulang selama waktu yang dibutuhkan dalam sehari.

25 menit betul-betul untuk bekerja, bukan membuka laptop tapi bermain social media, lalu setelah 25 menit ini selesai HARUS beristirahat selama 5 menit. Keluar ruangan, mengatur nafas, minum air putih dan memberikan mata sedikit olahraga dari sinar layar. Setelah beristirahat silahkan kembali dengan pekerjaannya. Teknik ini juga cocok untuk belajar.

# Bisikan Ngga Pantas Akan Datang

Kita selalu pindah dari satu ketakutan ke ketakutan yang lain. Waktu kecil kita takut hantu, takut suntik atau takut ngga naik kelas. Takutnya kita mungkin beda-beda saat kecil, satu hal yang pasti, kalau masih sempat baca tulisan ini, berarti kita masih hidup, masih bisa istirahat dan masih bisa makan saat bangun.

Lebih penting lagi, kita masih bisa berpikir langkah apa yang mau diambil untuk aktivitas berikutnya dan masih bisa belajar hal yang baru. Ketakutan waktu kecil tadi, udah ngga relevan sekarang.

Berbeda seperti saat masih kecil, sekarang takutnya beda lagi, kita takut gagal saat membuat usaha, takut ditolak saat melamar kerja (dan melamar wanita/pria) atau takut nilai jelek di kuliah. Semua rasa takut ini atau bentuk lainnya akan kita lewati. Percaya ngga percaya, ketakutan ini sebenarnya ngga seseram yang kita bayangkan.

Kita takut ngerjain satu hal, padahal kita sangat mau jadi sesuatu, kita mau mengubah sekitar jadi lebih baik, kita mau berkontribusi ke hal-hal besar, kita ngga mau jadi orang rata-rata. Tapi takut apa kata orang nanti, tapi takut apa yang terjadi setelah itu. Padahal semuanya ngga seseram yang kita bayangkan.

Hal yang paling buruk bisa terjadi adalah rencana kita “gagal”, tidak sesuai harapan dan tanpa sadar, kalau kita ngga ngelakuin sesuatu, alias sekarang ini, ya recanana kita udah ngga berhasil karena kita ngga ngelakuin apa-apa, jadi apa bedanya, kondisi sekarang dimana rencana kita belum terjadi, apa salahnya dicoba.

Dulu, saya takut menentukan jurusan kuliah, saya takut mau mengambil karir di bidang apa. Saat sudah belajar koding pun banyak pilihan, saya takut salah langkah, saya takut salah belajar. Ilmu yang saya punya sekarang, ngga mungkin saya dapatkan kalau diam ketakutan waktu itu. Terlepas dari takut, saya tetap belajar, tetap mengambil keputusan, ngga semua terpakai, tapi semuanya jadi bagian saya sekarang.

Saya takut harus berkarir di bidang apa, tapi saya senang sharing, saya mau kerjaan saya bermanfaat untuk orang lain. Terpikir kenapa ngga sharing ilmu yang saya punya? saya memutuskan waktu itu merekam apa yang saya tahu dan upload ke youtube, karena belum banyak tutorial programming yang bagus.

Padahal skill ini sangat penting jaman sekarang. Sayangnya saya takut komentar orang dan saya takut ada teman yang menertawakan.

Saya membuat sekolahkoding, orang-orang bisa fokus berkarya meluaskan manfaat dengan membuat aplikasi dengan belajar di sini. Walaupun saya takut, saat itu saya tetap melakukan sesuatu, saya tetap merekam video, saya tetap bikin kurikulum, saat awal semuanya masih kaku, setiap ada bunyi bel, telepon atau suara tetangga, saya harus berhenti merekam video, kadang putus asa, kadang malu kalau ketahuan sedang merekam. Walaupun saya takut, saya tetap melakukan sesuatu, ngga diam dan mengeluh.

Saya mau berkontribusi banyak lewat sekolahkoding dan terus bisa sharing sesuatu dengan gratis, jadi saya harus punya penghasilan sendiri, bisa menafkahi keluarga nanti dan bisa donasi ke proyek yang membantu orang lain, tanpa meminta-minta, tanpa membebankan banyak user-nya. Saya memutuskan untuk berhenti bekerja di kantor, saya memutuskan untuk ngga kerja biasa, saya takut orangtua saya ngga setuju, saya takut gagal.

Saya memutuskan bikin sistem “premium” di sekolahkoding. Saya mencari harga yang pas, yang dibandingkan gaji yang akan didapatkan ketika kerja nanti, cukup satu bulan aja, uangnya itu sudah bisa menutupi biaya premiumnya selama satu tahun. Artinya biaya ini ngga ada apa-apanya



dibandingkan dengan manfaat yang didapatkan, tapi memungkinkan saya bisa terus lanjut berkarya di sekolahkoding.

Saya takut apa kata membernya kalau ada premium, saya takut banyak yang protes, saya takut kalau usernya ngga mengerti tujuan saya, saya takut user ngga melihat manfaatnya. Saya takut kalau ngga laku. Walaupun saya takut, saya ngga menunggu bintang dari langit jatuh, atau mimpi diberitahu mana jalan yang benar. Saya kerjakan programnya dan tawarkan ke membernya.

Saya mau sharing ilmu yang bermanfaat dan bisa dipraktekkan sama teman-teman lain apapun bidangnya. Saya mau sharing lewat podcast dan juga blog. Saya takut ngga ada yang dengerin podcastnya, saya takut ngga ada yang baca blognya. Walaupun saya takut, saya tetap mengetik di komputer. Saya ngga diam, menunggu penerbit datang atau orang terkenal memperkenalkan saya, saya mulai menulis, tanpa izin siapapun.

Resikonya ngga seseram yang saya bayangkan, ketakutan saya sama sekali ngga beralasan. Poin di cerita ini tentu bukan tentang saya, tapi kamu! Punya rasa takut ngga bikin kita spesial, yang spesial kalau kita tetep melangkah setelah rasa takut datang.

Sekarang waktunya kamu melakukan apa yang mau dilakukan, tentunya kegiatan yang positif, kegiatan yang bisa bermanfaat untuk orang lain. Pasti akan banyak muncul rasa takut, bukan hanya satu. Solusinya bukan diam tapi kerjakan, rasa takut akan lewat, semuanya ngga seseram yang kita bayangkan.

Kita kebanyakan komentator yang takut dan yang diam dengan masalah. Takut dihina saat menawarkan solusi, takut ngga terpakai produk yang dibuat, takut orang ngga setuju dengan pendapatnya.

Terus kenapa? memang ngga semua orang akan setuju dengan kita, ngga semua orang suka dengan kita, ngga semua yang kita usahakan bisa berhasil, tapi kita ngga perlu semua itu. Kita ngga perlu semua orang setuju, kita ngga perlu semua usaha kita berhasil. Yang kita perlukan kerjakan dengan maksimal apa yang kita percaya.

Mengomentari orang jauh lebih gampang daripada berkarya, tapi rasa senang menjalankannya dan manfaat yang dihasilkan bukan dirasakan sama komentator tapi aktor. Hasilnya memang bukan kita yang menentukan, kita cukup berusaha dan berdoa.

Kalau kita ngga melakukan apa-apa, itu resiko yang paling besar. Kata Seth Godin “bermain aman” resikonya lebih besar dan berani ambil resiko jauh lebih aman.

Katanya (saya sih percaya) orang-orang menyesal bukan karena apa yang dilakukan, tapi karena apa yang ngga dilakukan.

# Susah itu Pertanda Bagus

Sebagian besar orang mengganti karir atau berhenti mengejar cita-citanya karena merasa susah. Padahal susah ini yang menjadi acuan, membedakan kita dari orang-orang lain. Semua orang akan menemukan susah, semua kegiatan akan ada bagian susah. Orang sering salah mengerti merasa ngga tahu “passion”nya dimana, atau ngga bisa konsentrasi di satu hal.

Padahal yang sering terjadi, dia berhenti saat bertemu dengan bagian susah. Semua orang mau akhirnya, tapi ngga banyak yang mau menjalani prosesnya. Berikutnya, kalau teman-teman bertemu dengan yang namanya susah, ini pertanda baik, anggap bertemu kolam air di tengah padang pasir.

Jangan panik, tuliskan dengan detail apa masalahnya, pecahkan masalah ini jadi bagian-bagian kecil dan selesaikan satu persatu dengan tenang. Tahu persis masalahnya apa, agar bisa mencari solusinya.

Orang panik ngga akan berpikir dengan tenang, ngga akan menemukan solusinya. Ingat, bagian yang susah ini yang menyaring orang-orang yang betul-betul layak mengejar cita-citanya dengan orang-orang yang sekedar mengikuti orang lain.

Saya masih ingat kejadian yang membuat saya pusing tujuh keliling. Saat itu sedang hadir di salah satu acara, iseng-iseng membuka website sekolahkoding dan tanpa diduga websitenya ngga bisa diakses. Tanpa berpikir panjang, saya langsung izin dan terburu-buru pulang ke rumah. Kurang lebih 30 menit saya habiskan untuk mencari kesalahannya apa. Dan ternyata sama sekali bukan kesalahan di kode tapi sertifikat SSL (agar website bisa diakses dengan https saya habis) dan yang perlu saya lakukan saat itu hanya mengupgradenya.

Berpikir dengan panik tidak akan menyelesaikan masalah. Lihat masalah sebagai pelajaran baru!

# Localhost Memang Aman

Ngga ada yang melihat, ngga ada yang menilai, ngga akan ada yang kritik, ngga akan ada yang menghina, selamat datang di localhost (saat program kamu masih di komputer pribadi belum dipublish ke mana-mana).

Sayangnya orang lain ada di luar sana, bukan di localhost. Kritik yang kita butuhkan ada di luar. Program yang kita buat dibutuhkan atau tidak, keputusannya ada di luar sana, bukan di localhost. Keputusan apakah program yang dibuat perlu diperbaiki atau tidak diperlukan sama sekali ada di luar sana.

Di luar tidak aman, tapi itu yang kita butuhkan, beli nama domain, cari tempat hosting, biarkan karyamu keluar dari localhost. Kritik akan datang, ini yang kita tunggu.

# Kamu akan lelah

Kalau kamu sedang merasa benar-benar lelah, ngga sanggup dan mau menyerah, ada kabar baik, itu normal! Silahkan beristirahat dan nanti kembali lagi. Belajar lagi, berkarya lagi.

Kamu lelah ngetik? tahu ngga masih ada orang yang bekerja jauh lebih keras dibanding kita secara fisik? Tangannya bisa sampai luka, bukan sekedar lelah. Kamu akan lelah, tapi bukan berarti lelah jadi alasan untuk berhenti.

Mau tahu cara cepat biar jago ngoding? Ambil air 3/4 gelas, siram ke muka, dan ingatkan diri sendiri kalau ngga ada yang instan. Okay kali ini saran asli, belajar dengan tenang dan punya tujuan yang jelas, bikin sesuatu dan lakukan ini berulang-ulang, berbulan-bulan bahkan bertahun-tahun.

# #Versi Terbaik Kamu



# Belajar Meneliti

Research atau meneliti, istilah yang cukup tegang, biasanya dipakai di laboratorium oleh para ilmuwan. Istilah keren ini juga bisa dipakai bahkan wajib di dunia programming.

Ada banyak hal yang bisa kita teliti, seperti teknologi apa yang cocok untuk masalah yang mau diselesaikan. Setiap bahasa program, framework atau library ada kelebihan dan kekurangannya, belajar meneliti, mencari referensi jadi bagian penting sebelum memulai.

Atau meneliti dengan melihat orang/perusahaan lain bagaimana menyelesaikan kasus yang mirip dengan masalah yang kita hadapi.

Ngga ada masalah, kalau kamu mau bereksperimen sendiri sebelumnya, terutama dalam proses belajar, perlu melatih logika sendiri dulu, sebelum asal mencari jawaban dengan instan.

Biasakan mencatat apa yang akan dicari (diteliti) karena kalau ngga, kita bisa dengan mudah terhisap di kegelapan internet. Internet menawarkan semua hal yang bisa bikin lupa tujuan awal kita datang.

Setelah melakukan penelitian, jangan lupa memikirkan hasilnya, untuk membawa kita satu langkah lebih kongkrit untuk menyelesaikan program berikutnya.

# Belajar Bahasa Inggris

Serius? Dua riuus! apapun pekerjaan yang pada akhirnya jadi pilihan kamu, bahasa inggris harus masuk daftar kurikulum wajib yang perlu pelajari. Mayoritas referensi belajar masih dalam bahasa inggris, termasuk programming.

Dokumentasi resmi dari suatu library atau framework juga normalnya dibuat dalam bahasa inggris. Meskipun ada translasi, seringkali pesan utama yang disampaikan oleh penulis (dokumentasi) akan hilang dengan terjemahan yang tidak akurat.

Bukan hanya dalam hal pasif seperti mengkonsumsi referensi bacaan atau video, tapi juga hal aktif, ketika kamu mau sharing, bertukar pikiran dengan orang lain, kenapa membatasi diri hanya dengan orang Indonesia? ada banyak orang di luar Indonesia, berkomunikasi, bertanya, membantu orang lain di luar Indonesia, perlu bahasa lain.

# Boleh deh, Tapi Free kan?

Boleh deh, tapi free kan? Pertanyaan yang ngga asing lagi di forum-forum.

Berapa banyak waktu dan uang yang dipakai untuk hiburan? menonton bioskop? makan makanan fast food? jalan-jalan? Kita jarang berpikir dua kali untuk hal-hal ini , tapi bagaimana dengan ilmu dan mengembangkan diri kita sendiri? Rasanya berat untuk mengeluarkan uang?

Sayangnya hiburan ngga bertahan lama. Seperti minum air laut, efeknya hanya menambah rasa haus. Cepat atau lambat teman-teman akan sadar investasi ilmu ke diri sendiri sangat penting , kalau sekarang berhenti belajar, sama saja menyerah dengan cita-cita kita dan bersiap menjadi rata-rata. Kalau mental kita selalu “boleh deh, tapi free kan?” , kita ngga akan maju.

Uang itu bisa dicari, uang itu alat, bukan tujuan. Jangan mengeluarkan alasan “saya masih mahasiswa, saya masih pelajar, saya ngga bisa kerja” Kata siapa? kamu bisa kerja, tapi kamu ngga ada usaha untuk kerja.

Ngga harus kerja skill, bisa juga non-skill, yang penting menghasilkan uang, untuk kamu berinvestasi lagi ke diri sendiri.

Bukan hanya investasi berupa ilmu, investasi berupa barang pun perlu dilakukan. Saya sampai sekarang tidak menyesal membeli “meja berdiri” meja agar saya bisa bekerja dalam posisi berdiri atau membeli ekstra monitor agar pekerjaan menjadi lebih produktif. Jangan melihat uang hanya nilainya saat ini, ada banyak hal yang bisa kita investasikan dengan uang.

“Waktu terbaik untuk investasi adalah di masa lalu” - unknown

# Berpikir Garis Besar

Sering kali kita ngoding hanya mengandalkan jari-jari, percaya semakin banyak mengetik semakin produktif. Padahal ngga sama sekali. Berpikirlah lebih banyak dan menulislah lebih sedikit.

Program yang ngga dipikirkan dengan baik, ngga akan berjalan dengan baik. Masalahnya akan mulai terlihat ketika kita mau mengubah dan mengupdate beberapa baris kode, akibat dari tidak merancang atau berpikir dengan baik akan membuat kita jauh lebih tidak produktif ke depannya.

Silahkan menggunakan kertas dan pulpen, ngga harus software mahal sama sekali. Merancang program secara garis besar, seperti blueprint dari apa yang akan kita buat. Fungsi (bukan nama fungsi, tapi hal apa yang bisa dilakukan di program ini) apa saja yang akan ada, struktur database seperti apa yang diperlukan, bagaimana hubungan antara satu data ke data yang lain.

Kalau bekerja dengan orang lain atau di perusahaan disarankan untuk menggunakan UML diagram sebagai “bahasa universal” yang bisa dimengerti orang lain.

Biasakan menyediakan waktu untuk berpikir di awal, sebelum memulai ngodingnya. Menjadi arsitek yang merancang bangunan kokoh, bukan bangunan yang ditiup angin jadi layu. Ingat kata orang bijak, penyesalan datang di akhir, kalau di awal namanya pendaftaran.

# Refactoring & Eliminasi

Merasa program kamu sudah selesai? “Salah! program ngga akan pernah selesai” Setiap program yang kita rasa sudah cukup bagus hari ini, besok masih bisa diperbaiki.

Refactoring, cukup sulit menemukan arti kata bahasa indonesia yang tepat untuk ini, yang jelas aktivitas ini digunakan untuk melihat kembali dan menulis ulang kode yang sudah berjalan, bukan menambah fungsi baru, dengan tujuan membuat kode kita lebih rapi, mempermudah mencari ketika ada error dan mudah menambahkan fungsi-fungsi baru nantinya.

“Kode yang indah ditulis, akan indah dibaca”. Hari ini mungkin benar programnya sudah selesai, tapi bagaimana kalau besok ada tambahan ide baru? atau client ingin menambahkan fitur lainnya? Jangan menyiksa diri, dengan menulis kode yang sulit dibaca. Bukan hanya ketika ada waktu, tapi perlu dijadwalkan untuk melakukan “refactor” untuk kode yang lama.



Bisa jadi ada kode yang punya fungsi sama tapi kita tulis berkali-kali, disini kita bisa membuatnya lebih efisien dengan membuat satu fungsi general, yang bisa diakses berkali-kali. Atau melihat ada satu fungsi yang melakukan banyak hal, sangat disarankan untuk setiap fungsi hanya melakukan satu hal saja, ngga lebih. Pecah fungsi tadi jadi beberapa bagian, yang tugasnya masing-masing jelas melakukan satu hal. Ada banyak lagi bentuk praktek dari refactoring.

Eliminasi ngga kalah penting atau secara kasar bisa kita artikan menghapus kodenya. Saya sering meninggalkan kode begitu saja, padahal sudah tidak terpakai, hanya dikomentari agar tidak ikut berjalan. Padahal kalau memang sudah tidak terpakai kenapa ngga dihapus? Semakin sedikit kode yang kita tulis, akan semakin mudah mengorganisirnya, akan semakin mudah menambahkan fitur-fitur baru nantinya.

Perhatikan berkali-kali kalau ada bagian dari kode kita yang tidak terpakai, berani mengeliminasi! Contoh lain seperti mengambil data semua kolom di table database, padahal tidak semua diperlukan. Ambil hanya yang diperlukan saja, berani eliminasi!

# Don't Repeat Yourself

Jangan mengulangi aktivitas yang tidak perlu diulangi!  
Jangan menulis ulang kode yang tidak perlu ditulis ulang!

Sebagian kecil sudah kita bahas di bagian “refactoring”, tapi masih ada yang lebih besar dari mantra DRY (don't repeat yourself) ini. Kalau teman-teman mau membuat galeri foto misalnya, pertanyaan pertama, kenapa harus membuat ulang? tentu saya setuju kalau tujuannya untuk belajar atau mencari solusi yang terbaik. Tapi kalau sudah ada library yang bisa membantu kita dengan performa yang baik, kenapa harus menulis ulang?

Pertimbangan waktu yang menjadi menu utama di sini. “Yang paling jauh adalah masa lalu” kata seorang imam. Ngga merhargai waktu sama seperti ngga menghargai diri sendiri. Hemat waktumu, dengan tidak menulis hal yang sama berkali-kali.

Belajar menulis dokumentasi untuk hal-hal yang akan kamu temui berkali-kali. Misalnya ada error yang sering muncul, kenapa ngga menulis error dan solusinya di satu file?

ketika kamu ketemu masalah yang sama, ada referensi yang cepat bisa kamu akses. Ini juga salah satu alasan kenapa saya membuat forum di sekolahkoding, agar semua orang punya referensi ketika bertemu error yang sama, tanpa perlu menghabiskan waktu yang lama. Alasan yang sama kenapa saya menulis buku ini, kenapa saya membuat video.

Contoh lain, ketika kamu harus mengirim email yang sama setiap 1 bulan satu kali, sudah ada sistem “scheduling” cron job yang bisa dimanfaatkan, kamu cukup menulis programnya satu kali dan akan otomatis terus dijalankan. Jangan mengulangi hal yang sama.

Cari hal-hal yang kamu terus ulang berkali-kali, jika bisa dibikin otomatis, bikin jadi otomatis! itu prinsip dasar dari programmer. Tulis semua hal yang kamu lakukan berkali-kali (hal yang sama) setiap hari atau setiap bulan, cari cara untuk membuatnya otomatis.

Ketika kamu mengerjakan proyek yang sama berkali-kali, seperti membuat profil perusahaan misalnya, kenapa harus selalu menulis ulang semuanya dari awal? kenapa ngga bikin satu template dasar, yang nanti bisa kamu kembangkan dan modifikasi sesuai kebutuhan klien nantinya. Jangan mengulangi hal yang sama.

“Jangan mengulangi hal yang sama.” agak miris memang saya mengulang-ulangi kata jangan mengulang.

# Skill Menyelesaikan Masalah

Konon katanya salah satu tujuan programmer diciptakan untuk menyelesaikan masalah yang ada di muka bumi ini.

Skill untuk menyelesaikan masalah perlu terus dilatih, sama seperti ketika kamu mau jago main bola, ngga ada cara lain selain bermain bola, bukan hanya dengan menonton video tutorial bagaimana bermain bola. Cara terbaik untuk melatih skill ini dengan terbiasa menyelesaikan masalah.

Saya masih ingat, tugas-tugas awal di programming, membuat fungsi untuk menebak bilangan itu ganjil atau genap, atau mengubah angka suhu dari celcius ke fahrenheit misalnya. Hal-hal dasar yang akan melatih “otot” menyelesaikan masalah kita.

Gabungkan dengan teknik “berpikir garis besar” sebelumnya, tahu persis masalah apa yang mau diselesaikan, seperti apa penyelesaiannya di bahasa manusia dan menerjemahkannya ke bahasa program.

Contoh latihan yang bisa kamu lakukan, jika kamu tinggal di kos-kosan, bayangkan kamu harus membuat sistem daftar pembayaran bulanan mahasiswa di kos ini. Gimana kamu akan merancang databasenya? gimana kamu akan membuat programnya untuk digunakan oleh pemilik kos nanti. Masalah yang ada di dunia nyata, pikirkan dan terjemahkan ke kode yang kamu tulis.

Jangan hanya berhayal dan keliling mencari lowongan kerja, berharap suatu hari nanti akan praktek di tempat kerja, praktek sekarang dan pekerjaan akan datang.

# Bicara & Bantu Orang Lain

Setelah cukup lama berenang di kolam programming, skill kamu akan bertambah, pengalaman kamu jelas lebih banyak dibanding orang yang baru mencelupkan kakinya di kolam ini. Orang-orang butuh bantuan kamu, jangan biarkan mereka kabur tidak jadi berenang apalagi sampai tenggelam.

Memang menakutkan untuk mulai membantu orang lain, tapi ingat, kita pun saat memulai dulu mendapat bantuan dari orang lain, entah itu di dunia maya atau di dunia nyata. Belajar untuk mengajar.

Ngga perlu membayangkan untuk langsung berbicara di atas panggung di depan ratusan orang, cukup cari satu orang yang tertarik di bidang ini, bantu dia. Satu hal yang menarik yang akan kamu temukan, dengan membantu orang lain, kamu akan belajar banyak hal baru. Hal-hal yang sebelumnya kamu ngga mengerti justru akan menjadi lebih jelas ketika kamu harus menjelaskannya ke orang lain.

Salah satu tips belajar yang luar biasa, kalau kamu ngga bisa menjelaskan dengan mudah ke orang lain, berarti kamu belum mengerti. Jadi bayangkan saat kamu belajar sesuatu yang baru, bagaimana kamu akan menjelaskannya ke orang lain.

Selain dengan mencari satu orang tadi, kamu bisa mulai belajar menulis blog tentang yang kamu pelajari, membuat video di youtube atau mencari forum-forum tempat kamu bisa membantu banyak orang yang ngga kamu kenal di dunia nyata.

Mulai cari di internet dimana ada komunitas programming di kota kamu dan datang ketika ada acara. Gimana kalau ngga ada? bikin! cari satu dua orang yang tertarik, mulai dari pertemuan kecil. Dan pelan-pelan kembangkan komunitasmu, yap, seperti walikota, kamu akan punya pengaruh besar yang baik di daerahmu.

Komunitas programmer tidak akan sebesar sekarang, kalau masing-masing hanya mengurus dirinya atau semua malu ngga ada yang berani berbicara, takut disebut guru, takut disebut master. Ada banyak hal lain yang perlu kita takuti dibanding mendapat pujian atau sindiran.

# #Rezeki di Dunia Koding



# Butuh Sarjana & Sertifikat

“Apakah saya butuh menjadi sarjana atau mengumpulkan sertifikat untuk mendapatkan pekerjaan di dunia ini?”

TIDAKKKKKKKKKKKKKKK

# Siapkan Portofoliomu

Okay, bagian sebelumnya ngga banyak membantu, tapi itu pendapat jujur saya. Ini satu hal yang mungkin membantu, bikin portofolio!

Mungkin kamu ngga punya gelar atau sertifikat dengan nilai A+ yang bisa ditunjukkan, tapi bukan berarti kamu ngga bisa punya karya! Bikin sesuatu sekarang, bukan nanti saat ada lowongan kerja. Bikin hal yang luar biasa yang perusahaan ngga akan bisa menolak kamu hanya karena ngga punya kertas nilai atau tambahan 3-4 huruf di belakang namamu.

Sangat normal saat diawal karir di dunia kreatif, orang lain belum berani membayar untuk skill kamu. Ngga semua orang setuju, tapi saya sangat menyarankan untuk di awal, berani bekerja tanpa uang atau dengan harga yang murah. Pengalaman dan portofolio yang menjadi target.

Kalau keluarga kamu punya usaha, tawarkan untuk membuat website, jauh lebih baik, ketika kamu bisa deal dengan perjanjian jika website ini membantu usahanya,

kamu akan dapat bayaran nanti. Tapi sekali lagi ini bukan target utama di awal. Tapi jangan menjadikan ini kebiasaan, bekerja terus dengan gratis, ngga akan bisa membantu membayar kebutuhan kamu apalagi anak dan istri nanti. Bekerja dengan gratis tapi jadi meminta-minta jelas bukan pilihan.

Kalau tidak ada di keluarga yang bisa dibuatkan website, cari organisasi non-profit di kota kamu, untuk membantu publikasinya, tawarkan untuk membuat website, aplikasi atau program yang mereka butuhkan.

Sudah ada banyak tempat hosting gratis untuk menampilkan karyamu berupa website yang bisa diakses oleh orang lain, ngga boleh lagi ada alasan “saya ngga punya uang”. Kumpulkan karyamu, tuliskan pendapat dan komentarmu tentang proyek ini, apa tantangannya, apa yang kamu harapkan bisa diubah atau yang bisa kamu perbaiki. Bukan hanya komentar positif, karena karya yang jujur akan lebih dihargai orang lain. Ingat pendapat kamu tentang karya itu sendiri penting.

# Bekerja Tetap atau Sebagai Freelancer

Ada banyak pintu rezeki, sayangnya kita malas membukanya. Termasuk di dunia programming, ada berbagai macam pintu yang bisa kamu buka.

Kamu bisa mulai mencari perusahaan, melihat lowongan pekerjaan yang tersedia dan memperhatikan syarat-syaratnya. Latih skillmu sesuai syarat-syarat tadi tunjukkan dengan karya (portofolio, yang kita bahas di bagian sebelumnya) dan berani untuk melamar pekerjaannya.

Saat bekerja pun jangan berharap kamu sudah menguasai semuanya. Akan ada tantangan dan hal baru yang perlu kamu pelajari. Saat bekerja di kantor pertama, saya diminta untuk membuat aplikasi mobile ios, padahal saya melamar sebagai web developer.

Di saat seperti ini ada dua kemungkinan, kamu bilang ngga bisa dan minta mencari orang lain, atau kalau kamu suka tantangan dan belajar hal baru, seperti yang saya lakukan, terima dan berikan tawaran untuk memberi waktu belajar.

Selama kita memberikan alasan yang jelas dan jujur, orang akan mengerti.

Pilihan lain, sebagai freelance atau pekerja lepas. Tidak terikat di satu perusahaan, tapi hanya mengerjakan berdasarkan proyek dalam jangka waktu tertentu. Sangat cocok untuk kamu yang mudah bosan. Ada banyak website yang menawarkan pekerjaan freelance seperti ini.

Dua tipe pekerjaan di atas tidak mengharuskan kamu bekerja di satu tempat. Ada pilihan yang namanya “remote working” alias kamu bebas bekerja dari mana saja, bahkan lintas negara. Saya yakin ini udah ngga asing, kalau memang kamu merasa di kotamu pekerjaan terbatas, jangan hanya datang ke ibukota, ada opsi lain yaitu bekerja dengan perusahaan dari negara lain. Disinilah kenapa skill bahasa inggris ngga boleh kamu lewatkan. Kebutuhan sehari-hari menggunakan mata uang rupiah, tapi berpenghasilan dollar kan bukan tawaran yang buruk.

Cari di internet “remote programming job” akan ada banyak website yang menawarkan. Pastikan kamu punya koneksi internet yang baik, agar mudah berkomunikasi nantinya.

Masih ada? masih ada! kamu bisa juga membuat agensi atau jasa membuat program. Bsukan lagi kamu mencari lowongan pekerjaan, tapi orang-orang yang akan mencari jasa kamu. Tentu pada saat di awal kamu perlu mencari beberapa klien,

tapi dengan pengalaman dan kerja yang memuaskan, orang lain akan mendengar keahlianmu.

Sedikit catatan tentang bekerja dengan perusahaan atau orang lain:

jangan melanggar prinsip hidup kamu! jangan sampai hanya karena uang kamu rela menerima kerja apa saja, usaha yang merugikan orang lain atau sekedar sia-sia. Seperti bekerja di website film bajakan atau perjudian. Kamu punya prinsip, hargai dirimu sendiri.

# Programmer Bikin Bisnis Sendiri?

Masih ingat saya bilang banyak pintu rezeki yang ngga kita buka? ini salah satunya, bikin bisnismu sendiri! atau istilah yang lebih populer sekarang, startup. Ngoding sendiri memang ngga gampang, apalagi ditambah bisnis. Kamu harus siap menemukan banyak pekerjaan ekstra yang perlu dilakukan kalau memang mau membuat bisnis sendiri. Ngga gampang, tapi bukan ngga mungkin.

Butuh satu buku sendiri untuk membahas bagaimana membuat bisnis, saya hanya akan menjelaskan beberapa poin pentingnya disini. Meskipun bukan ahli di bidang bisnis, tapi setelah menjalankan sekolahkoding beberapa lama, saya sadar, tanpa investasi miliaran, tanpa kantor yang besar, tanpa endorsement dari artis yang populer, kamu tetap bisa menjalankan bisnismu sendiri.

Kamu bisa mulai dari ide atau dari melihat masalah yang ada dan memberikan solusi ke masalah ini. Ide? orang lain belum tentu butuh. Masalah? sudah jelas ada yang butuh. Bisa jadi solusimu menghemat waktu orang lain,

menghemat uang dan masih banyak lagi nilai-nilai yang bisa kamu kontribusikan.

Kamu bikin sesuatu yang “wah”, ngga akan ada yang tahu sampai kamu umumkan. Kalau kamu bertapa 3 bulan bikin program tapi hanya di localhost, ngga akan ada yang tahu. Berani publish di social media dan minta pendapat orang lain. Bukan hanya pendapat teman atau keluarga, karena orang-orang yang dekat dengan kita hanya akan memuji untuk membuat kita senang. Alternatifnya cari tahu siapa target konsumen kamu, untuk siapa bisnismu ini. Kamu bisa cari di dunia maya atau di dunia nyata. Tanyakan pendapat mereka tentang produk kamu, pendapat yang jujur orang-orang ini.

Fakta menariknya kamu ngga harus sendiri. Kamu bisa cari teman seperjalanan, partner yang rela ngga dibayar, mau berjuang bersama di awal. Saling mengisi kekurangan, jangan cari partner yang semuanya programmer, cari designer, cari orang bisnis, cari orang marketing, cari ahli copywriting. Saling mengisi kekurangan.

Tapi kalau mereka ngga dibayar mana mungkin mau bergabung? Mungkin! ceritakan apa visi misi kamu, kenapa bikin bisnis ini, masalah apa yang mau diselesaikan. Siapa yang tahu kalau mereka memang tertarik dan peduli dengan masalah ini. Urusan uang kamu bisa membagi saham di awal, kalau nanti bisnisnya sudah jalan.



Yang sering dipraktekkan juga, orang-orang tetap bekerja di suatu perusahaan atau membuat jasa pembuatan program sambil menjadikan bisnisnya ini pekerjaan sampingan, agar tetap menghasilkan.

Selama ini kita punya teman untuk olahraga, ada teman bermain game, dan hobi-hobi lain. Kenapa hanya membatasi diri disini? kenapa ngga nyari teman yang sama-sama mau berubah dan berjuang.

Mulai dari hal yang kecil, jangan menggabungkan banyak hal menjadi satu, semakin banyak yang mau dikerjakan, semakin banyak beban yang akan datang. Menyerah di jalan, proyeknya terbengkalai, hilang semangat dan lain-lain. Fokus ke satu hal, fokus menyelesaikan satu masalah. Eliminasi bagian yang tidak penting.

Waktu saya membuat sekolahkoding, saya ngga punya forum, saya ngga punya blog. Inti dari sekolahkoding hanya video tutorial. Dan untuk membuktikan ini, cukup upload ke youtube, apakah orang lain mengerti dan senang dengan tutorial yang saya buat. Ngga perlu bikin website, ngga perlu bikin blog, ngga perlu bikin forum di awal.

Forum di sekolahkoding sendiri saya buat setelah ngga bisa menjawab semua pertanyaan satu per satu di berbagai video yang ada di youtube. Kenapa ngga bikin satu tempat di mana orang bisa saling berdiskusi dan membantu satu sama lain? ini contoh lain membuat sesuatu dari masalah.

Masih banyak diantara kita yang merasa “semakin banyak semakin bagus”. Padahal semakin banyak semakin repot. Cari satu hal, inti dari produk yang mau kamu buat, eksekusi ini dengan sangat baik, lempar ke publik, lihat tanggapan yang jujur dan perbaiki dari sini, terus ulangi siklus ini. Jangan khawatir nanti kamu bisa terus merubah atau menambahkan produkmu kalau memang diperlukan.

Tidak seperti dongeng, ngga semua berakhir indah. Jangan berharap produk pertama kita akan langsung berhasil. Semua ada prosesnya, kamu akan belajar dari satu kegagalan ke kegagalan yang lain. Jangan marah kalau produk kamu ngga diterima, sebaliknya tanyakan kenapa ngga diterima, memang produknya ngga diperlukan? atau eksekusinya yang kurang baik? jujur dengan diri sendiri, bukan dengan menebak-nebak tapi menanyakan ke penggunanya langsung.

“Semua langkah pertama berat, tapi semua perjalanan di mulai dari langkah pertama”

# #Sekarang Giliran Kamu

# Tool yang Paling Bagus

Saya ingat kisah lucu (tidak asli) saat J.K. Rowling, penulis Harry Potter mengadakan jumpa fans. Dan fans-fans ini diberi kesempatan untuk bertanya. Ada satu yang angkat tangan, “Mba J.K., saya fans berat Harry Potter! keren banget tulisannya, pertanyaan saya, nulisnya pake pensil apa?”

Apakah kamu setuju kalau pertanyaan dari fans ini ngga penting? Bukan pensil apa yang dipake untuk menulis novel Harry Potter jadi luar biasa, tapi ceritanya. Setiap kali kamu pusing mau berkarya pakai apa, tool apa yang paling bagus, ingat karya kamu yang penting, bukan toolnya.

# Kode yang Sempurna

Ngga akan datang hari dimana kamu ngerasa kode yang ditulis sudah sempurna. Untungnya berkarya tidak perlu sempurna. Jadi jangan menunggu sempurna. Error bisa diperbaiki, kesalahan bisa dipelajari, tapi pengalaman ngga akan datang sendiri.

Berkarya yang terbaik hari ini, kamu di masa depan akan berterima kasih dengan kamu sekarang, yang terus berkarya, terus belajar dari kesalahan, ngga takut salah, ngga nunggu hari terbaik datang, ngga nunggu kesempatan emas hadir. Pancing kesempatan emas dengan terus berkarya setiap hari.

Programmer (Kamu) punya keuntungan yang luar biasa dibanding orang lain, untuk berkarya orang lain membutuhkan banyak hal, butuh alat, butuh modal besar dan lain-lain. Kamu hanya perlu menyalakan komputer dan mulai menulis programnya.

Bayangkan kalau yang bikin facebook berpikir dia ngga punya cukup sertifikat programming saat ini, dia belum punya gelar dari kampusnya dan karena itu dia berhenti membuat facebook. Jangan sampai hal terbaik yang akan hadir berhenti hanya karena kamu nunggu sesuatu yang tidak perlu ditunggu.

# Sapi Warna Ungu

Buku yang ngga berhenti saya rekomendasikan adalah “purple cow” dari Seth Godin.

Bayangkan kamu sedang pulang kampung, ada banyak sapi warna coklat di ladang, ngga ada yang spesial. Tapi bayangkan kalau kamu melihat sapi warna ungu. Saya sih akan mampir, memotret dan memperhatikan sapi ini dengan baik. Sapi warna coklat ngga menarik karena sudah banyak, sapi warna ungu yang menarik.

Ada banyak programmer, tapi apakah kamu programmer warna ungu? punya warna yang beda, punya karya yang unik, yang bisa membuat kamu lebih menarik dibanding programmer lain?

# Apa Hadiahmu?

Dunia sudah punya cukup banyak masalah  
Dunia kekurangan yang mau menyelesaikan masalah.

Kamu bisa melihat dirimu sebagai pahlawan, yang ngga peduli dengan namanya akan diingat atau ngga, tapi peduli dengan masalah yang ada dan mau menyelesaikannya sekarang.

Kamu bisa melihat dirimu sebagai seniman, seseorang yang akan dinikmati karyanya oleh banyak orang. Persembahkan karya terbaikmu. Komputermu sebagai panggung, kamu sebagai pemainnya, mulai pentasnya sekarang. Semua orang sudah menunggu.

Ngga ada masalah kalau kamu memutuskan untuk ngga jadi programmer saat ini atau jadi programmer seumur hidup. Jujur saja, saya juga ngga yakin akan selamanya jadi programmer, tapi saya ngga akan pernah menyesal sudah menjadi programmer. Saya punya kekuatan super untuk menyelesaikan masalah yang ada di dunia ini.

Ini dia hadiah dari saya.

Sekarang apa hadiahmu untuk orang lain?



# Terima Kasih!

Hey! semoga buku ini bisa jadi teman perjalanan kamu di dunia programming. Tolong berikan Komentar/kritik kamu tentang buku ini [disini](#) atau mention saya langsung di [@hilmanspace](#)

Praktekkan apa yang bermanfaat dan tinggalkan apa yang ngga. Social-media saya atau email? [hilman.space](#)

Postingan-postingan kecil lainnya bisa kamu lihat di [Instagram SekolahKoding](#) .

Kalau memang buku ini membantu kamu, besar kemungkinan akan membantu banyak orang lain.

Bantu saya untuk menyebarkannya ke temanmu, ajak orang lain untuk membaca tulisan ini.

[Link Buku Hadiah Untuk Programmer](#)