

GUIDELINE SKILL REVIEW

ASSESSMENT

1. Pengumpulan bukti untuk assesment

Tujuannya pengumpulan bukti ini adalah untuk mempermudah asesor (Lead Developer/Engineer Manager) yang tidak bekerja secara day to day tetap mendapatkan bukti bukti yang relevan untuk melakukan judgement skill yang sudah kamu lakukan dengan baik. selain itu, bukti assessment tersebut juga membantu developer untuk “menunjukkan” ke asesor bahwa anda “sudah mampu” melakukan keterampilan pada tingkat tertentu.

2. Jenis Jenis Bukti yang dikumpulkan Dan Proses pengumpulan bukti

Jenis bukti yang bisa dikumpulkan dapat berupa atau gabungan dari :

- Menarasikan dengan dokumentasi spesifik

Anda dapat menggunakan kata kata sendiri menggunakan dokumentasi apapun untuk menjelaskan bagaimana anda telah menerapkan skill tersebut di berbagai situasi. Lengkapi narasi yang anda buat dengan memilih berbagai bukti seperti Slack diskusi, GIT, pull request , Metrik yang berada di sistem yang mendemonstrasikan achievement (misal coverage unit test 100%) dan lain lain selama 6 bulan terakhir.

- Presentasi Live Coding

Anda juga bisa mempresentasikan kembali proses development yang pernah anda buat. pilih berbagai proses development yang relevan sesuai dengan aspek aspek skill yang ingin kamu tunjukkan selama 6 bulan terakhir ini.

Jangan bergantung sepenuhnya pada satu jenis bukti. Developer bisa menggabungkan berbagai jenis bukti yang relevan untuk memperkuat argumentasi.

Tips penulisan bukti dan konten bukti

- Susun jawabanmu dengan efektif seperti penggunaan PEE (Point, explanation, Evidence) atau STAR Method (Situation, Task, Action, Result)
- Jangan menggunakan font yang aneh yang sulit dibaca
- Gunakan Bullet point jika diperlukan
- Gunakan grammar yang tepat, gunakan dengan kalimat aktif

3. Proses Assessment

Setelah developer mengumpulkan bukti bukti dari berbagai aspek, selanjutnya akan di jadwalkan untuk 1on1 bersama Asesor (Lead/Engineer Manager) oleh HR.

Developer akan melakukan presentasi dari bukti bukti tersebut pada 1on1 session kepada assesor, dan asesor akan melakukan penilaian dari bukti bukti tersebut (dan mungkin ada tanya jawab).

4. Skill Apa yang harus dibuktikan

Skill Terdiri dari 10 Aspek yaitu Communication, Documentation, Debugging & Technical Problem Solving, Planning Delivery & Estimation, Technical Quality, Language Knowledge, Tooling, performance, security dan Focused Technical (Frontend Mobile, dan Backend).

Developer harus mengumpulkan bukti dari 9 Aspek skill tersebut saat melakukan skill review, untuk komunikasi tidak dikumpulkan bukti tapi berdasarkan feedback dari klien.

5. Proses Promosi ke Level Selanjutnya

Proses skill review dimulai dengan level kondisi engineer saat ini dan harus memuaskan dari semua aspeknya pada level tersebut sebelum naik ke level selanjutnya. Misalnya jika engineer pada level Junior developer ingin naik level ke level medior developer, dia harus bisa memuaskan semua aspek junior terlebih dahulu walaupun ada aspek yang sudah masuk syarat pada level medior dan begitu juga seterusnya. tujuannya agar developer tersebut memastikan aspek aspek yang belum terpenuhi ke syarat medior dapat memperbaiki skillnya yang belum terpenuhi terlebih dahulu.

LEVEL DEVELOPER

Skyshi mendukung perkembangan kemampuan setiap developer yang didefinisikan melalui dari level developer yaitu Junior, Medior dan Senior, dari point point dibawah ini kamu akan mengetahui ekspektasi seperti apa level developer yang sesuai dengan ekspektasi skyshi

1. Junior Developer

Seorang junior developer memiliki skill yang cukup untuk mulai memberikan “value” ke klien atau project. Mereka memiliki pemahaman fundamental bahasa pemrograman, mampu mengikuti prosedur dan koding standar yang telah diberikan dan proaktif berkomunikasi secara rutin dan transparan dengan menggunakan berbagai macam komunikasi platform kepada tim atau klien.

Junior developer juga fokus untuk inisiatif terjun ke berbagai jenis project untuk memperbanyak pengalaman dan memperdalam kemampuannya dalam memahami suatu produk / flow requirement dengan bimbingan dan arahan dari senior developer.

2. Medior Developer

Seorang medior developer memiliki landasan teknis yang cukup luas dan sudah mampu bertanggung jawab dalam menghasilkan dan mempertimbangkan kualitas kode yang baik, memiliki pengetahuan terhadap bahasa yang sudah mempertimbangkan best practice dalam penggunaannya dan konsisten berkomunikasi secara profesional dengan klien.

Seorang medior developer masih perlu menambah pengalaman untuk meningkatkan pengetahuan teknis secara mendalam untuk mendapat lebih banyak solusi - solusi yang efektif.

3. Senior Developer 1

Seorang Senior Developer 1 mampu deliver task yang kompleks secara menyeluruh dengan penerapan best practice yang baik. Mereka memiliki penguasaan bahasa dan pemahaman teknis yang mendalam, selalu konsisten memastikan kualitas kode yang dibuat memiliki kualitas yang baik sehingga mampu bertanggung jawab pada sistem yang lebih besar dan memiliki kemampuan komunikasi yang menakjubkan.

Senior Developer 1 juga sudah memiliki banyak pengalaman yang membuatnya mampu mengartikulasi dan memahami berbagai kebutuhan/requirement dengan mudah dan memberikan solusi yang efektif.

4. Senior Developer 2

Senior Developer 2 memiliki pemahaman teknikal yang mendalam dengan banyak pengalaman dalam penggunaan berbagai macam bahasa pemrograman, framework dan berpengalaman dengan berbagai jenis produk domain/platform. Mereka mampu secara efektif dan bertanggung jawab untuk memutuskan kebutuhan teknis dan software arsitekturnya seperti apa dan memiliki kemampuan problem solving yang exceptional.

Dengan knowledge yang luas dan exceptional tersebut, senior developer 2 mampu melakukan review pekerjaan engineer lain.

SKILLS

COMMUNICATION

Definisi :

Kemampuan dalam transparansi informasi yang dilakukan secara day to day, efektifitas dalam berkomunikasi dan kemampuan dalam menginformasikan permasalahan teknis serta hasil pekerjaan.

Junior Developer

- Mampu rutin dalam menuliskan to do list dan report di channel komunikasi dengan tim maupun klien
- Mampu responsif saat berkomunikasi dengan tim
- Mampu mengikuti meeting secara reguler
- Mampu proaktif dalam mencari feedback jika mendapatkan kesulitan saat development
- Mampu menggunakan fitur fitur advance di berbagai communication tools, seperti telegram, slack, Whatsapp dan lain lain
- Mampu menggunakan fitur fitur advance di berbagai voice/video calls communication tools seperti Zoom dan google meet

Medior Developer

- Mampu percaya diri berkomunikasi dengan klien,
- Mampu berkomunikasi dengan cara penyampaian yang lugas, straight forward dan tidak berbelit belit kepada kolega atau klien
- Mampu menyampaikan ide-ide dengan baik dan mudah dipahami orang lain
- Mampu menyampaikan feedback secara jelas melalui commit messages, Jira Comment atau PR Deskripsi dll
- Mampu memberikan feedback secara rutin kepada klien atau kolega.
- Mampu menjaga komunikasi tetap berjalan baik dengan klien atau tim untuk meminimalisir miskomunikasi

Senior Developer 1, 2

- Mampu mengambil inisiatif untuk memimpin meeting
- Mampu melakukan presentasi atau demo yang excellent/menakjubkan dengan menggunakan berbagai tools presentasi
- Mampu menjembatani komunikasi antar tim untuk membantu dalam pengambilan keputusan dalam memecahkan masalah.
- Mampu menjadi mediator komunikasi antar tim / klien jika terjadi perdebatan yang mungkin terjadi kebuntuan/ deadlock.

DOCUMENTATION

Definisi :

Kemampuan dalam mendokumentasikan untuk menyediakan informasi/knowledge secara teknis ataupun non teknis kepada developer lain atau klien

Junior Developer

- Mampu mengikuti dan membaca standar dokumentasi yang sudah dibuat oleh developer lain
- Mampu membaca dokumen deployment yang sudah dibuat oleh developer lain

Medior Developer

- Backend
 - Mampu mendokumentasikan rencana teknis seperti pengembangan API atau database seperti dokumentasi di Postman custom feature dll
 - Mampu mendokumentasikan ERD jika mengerjakan di awal project
 - Mampu mendokumentasikan dokumen Deployment
 - Mampu menulis dokumentasi non-teknis, misalnya panduan pengguna untuk fitur khusus
 - Mampu menyediakan konteks untuk pull requests termasuk deskripsinya, dan screenshot, dan eksternal link yang relevan

- Mampu mendokumentasikan kode secara proper beserta fitur fitur teknis jika diperlukan
- Frontend dan Mobile
 - Mampu mendokumentasikan rencana teknis seperti halaman , integrasi API yang diakses, user guide atau custom feature dll

Senior Developer 1, 2

- Mampu menulis dokumentasi yang jelas, ringkas dan tepat mengcover seluruh aspek yang dikerjakan
- Mampu berkontribusi pada developer guideline seperti konsisten melakukan update dokumentasi dan mendorong developer untuk menggunakannya
- Berkontribusi pada PR template, memastikan konten PR tersebut relevan dan digunakan oleh tim
- Mampu memperkenalkan dan mendefinisikan template PR, mengedukasi tim bagaimana cara penggunaanya dan mendorong kolaborasi perubahan template

PLANNING, DELIVERY & ESTIMATION

Definisi :

Kemampuan dalam melakukan perencanaan task yang harus dikerjakan, melakukan spesifikasi dan analisis task, melakukan breakdown task dan melakukan estimasi task berdasarkan level kompleksitas sebuah flow/business requirement

Junior Developer

- Mampu mengimplementasikan task yang sudah diberikan oleh senior developer dengan baik
- Mampu memprioritaskan task yang sudah breakdown oleh senior/lead developer dengan baik
- Mampu memahami bagaimana user berinteraksi dengan product
- Mampu mengestimasi task task yang kompleksitasnya simple dengan akurasi yang reasonable

Medior Developer

- Mampu mengestimasi task task yang kompleksitasnya medium dengan akurasi yang reasonable
- Mampu memahami, menganalisis dan menginterpretasikan flow requirement fitur fitur yang sifatnya cukup kompleks dan mem break down requirement tersebut dengan baik.

Senior Developer 1, 2

- Mampu mengestimasi task yang kompleksitasnya rumit dengan akurasi yang reasonable
- Mampu memahami, menganalisis dan menginterpretasikan flow requirement yang kompleks dan mem break down requirement tersebut dengan baik
- Mampu mendelegasikan hasil interpretasi flow requirement kepada junior developer,
- Mampu membantu menjelaskan flow requirement kepada tim dan/atau klien
- Mampu memberikan alternatif flow requirement kepada klien,

DEBUGGING & TECHNICAL PROBLEM SOLVING

Definisi :

Kemampuan dalam mencari dan menentukan akar penyebab masalah teknis dan penerapan solusi yang tepat.

Junior Developer

- Mampu menginvestigasi dan fixing basic issues, dan masih perlu pendampingan/bantuan dari senior developer
- Mampu menduplikasikan issue dari laporan issue yang dilaporkan oleh developer lain atau QA

Medior Developer

- Mampu memahami cara memasang log error.
- Mampu melakukan debugging seperti membaca melalui error log atau menggunakan step-through debugging
- Mampu menggunakan tools debugging yang sesuai/relevan saat melakukan debugging code

- Mampu menggunakan pendekatan sistematis untuk debug pada single service issue

Senior Developer 1

- Mampu melakukan debugging end to end issue yang kompleks,
- Mampu mendiagnosis secara sistematis dan mempersempit penyebab dengan proses eliminasi
- Mampu menggunakan pendekatan sistematis untuk debug pada cross services issue
- Mampu membantu developer lain untuk solving problem dan membuat mereka menjadi lebih mandiri dalam melakukan debugging
- Mampu berkolaborasi dengan QA/ bug reporters untuk mencoba mempersempit masalah yang sulit di reproduksi

Senior Developer 2

- Mampu melakukan debugging pada semua issue dengan waktu yang cukup
- Mampu mengetahui bagaimana me mitigasi / mengurangi masalah dimana masalah tersebut tidak bisa/belum ditemukan di waktu yang tepat
- Mampu menetapkan pendekatan sistematis untuk mendiagnosis issue issue yang tidak bisa direplikasi
- Mampu membantu developer lain saat melakukan debugging pada issue yang sangat kritikal / sulit

LANGUAGE KNOWLEDGE

Definisi :

Kemampuan mengetahui, memahami dan penguasaan mendalam terhadap bahasa pemrograman dan framework yang digunakan.

Junior Developer

- Frontend (React)
 - Menguasai fundamental bahasa pemrograman Javascript dan framework react dan penggunaannya pada common issues
 - Mampu menerapkan fitur fitur dasar React seperti Create React app, JSX, Component, List and Key, useState, useEffect, Props vs State, dll (Jika Menggunakan React),
- Frontend (Vue)
 - Menguasai fundamental bahasa pemrograman Javascript dan framework vue dan penggunaannya pada common issues
 - Mampu menerapkan fitur fitur dasar Vue seperti App Configuration, Rendering List, Conditional rendering, Lifecycle Hooks, Forms Handling, Events Handling, Computed properties, Vue CLI, Component, Template API styles dll
- Mobile (React native)
 - Menguasai fundamental bahasa pemrograman Javascript dan framework react native dan penggunaannya pada common issues
 - Mampu menerapkan fitur fitur dasar flutter basic seperti Basic styling widget, Stateless/Stateful widget, Inherited widget

- Mobile (Flutter)
 - Menguasai fundamental bahasa pemrograman Flutter dan penggunaannya pada common issues
 - Mampu menerapkan fitur fitur dasar flutter basic seperti Basic styling widget, Stateless/Stateful widget, Inherited widget
- Backend (Node Js)
 - Menguasai fundamental bahasa pemrograman Node.js dan penggunaannya pada common issues
 - Mampu menerapkan fitur fitur dasar Node Js seperti Callback, Event emitter REST buffer modul system dan lain lain (Jika menggunakan Node JS)
 - Mampu menerapkan fitur fitur dasar GO seperti Basic syntax, variables and declaration, Data Type, Conditionals, multiples/named return, Type casting, Type inference dan lain lain (Jika bahas utama menggunakan GO)
- Backend (GO)
 - Menguasai fundamental bahasa pemrograman Go dan penggunaannya pada common issues
 - Mampu menerapkan fitur fitur dasar GO seperti Basic syntax, variables and declaration, Data Type, Conditionals, multiples/named return, Type casting, Type inference dan lain lain (Jika bahas utama menggunakan GO)

Medior Developer

- Frontend (React)
 - Mengetahui sebagian besar seluk- beluk react js, dan bagaimana penggunaannya
 - Mampu menerapkan fitur fitur advanced React seperti Hooks, Context Ref, Render Props, High order component, dll
 - Mampu menerapkan ecosystem react seperti Forms, Testing, State management (redux), Routers, SSR SSG, API Calls dan lain lain
 - Memiliki pengetahuan yang cukup baik dalam menerapkan best practice dari fitur fitur react

- Mampu proaktif mempelajari fitur fitur terbaru /terupdate dari Reactjs
- Frontend (Vue)
 - Mengetahui sebagian besar seluk- beluk Vue js, dan bagaimana penggunaannya
 - Mampu menerapkan fitur fitur advanced Vue seperti Async component, Teleport Component, provide / Inject, Custom Directives, Custom Event, Plugins, Watcher, Slots dan lain lain
 - Mampu menerapkan ecosystem Vue seperti Vue Router, Forms, SSR, SSG, State management (Pinia), API Calls, Testing dan lain lain
 - Memiliki pengetahuan yang cukup baik dalam menerapkan best practice dari fitur fitur Vue
 - Mampu proaktif mempelajari fitur fitur terbaru /terupdate dari Vue.js
- Mobile (React native)
 - Mengetahui sebagian besar seluk- beluk react native, dan bagaimana penggunaannya
 - Mampu menerapkan fitur fitur advanced React seperti Hooks, Context Ref, Render Props, High order component, dll
 - Mampu menerapkan ecosystem react seperti Forms, Testing, State management (redux), Routers, SSR SSG, API Calls dan lain lain
 - Memiliki pengetahuan yang cukup baik dalam menerapkan best practice dari fitur fitur react
 - Mampu proaktif mempelajari fitur fitur terbaru /terupdate dari React native
- Mobile (Flutter)
 - Mengetahui sebagian besar seluk- beluk flutter, dan bagaimana penggunaannya
 - Mampu menerapkan fitur fitur advanced flutter
 - Mampu menerapkan ecosystem flutter
 - Memiliki pengetahuan yang cukup baik dalam menerapkan best practice dari fitur fitur Flutter
 - Mampu proaktif mempelajari fitur fitur terbaru /terupdate dari Flutter
- Backend (Node js)
 - Mengetahui sebagian besar seluk- beluk Node.js, dan bagaimana penggunaannya

- Mampu menerapkan fitur fitur advanced Node js Template literal, Multiline strings, Arrow function, Enhanced object literals dll
- Mampu menerapkan web framework seperti adonis.js, express.js, sails.js dan lain lain sesuai dengan kebutuhan
- Mampu menerapkan Log framework seperti Node-loggly, winston, Morgan dan lain lain sesuai dengan kebutuhan
- Mampu menerapkan Log management system seperti Loggly.com atau sentry.io
- Mampu menerapkan Template engine dan Real time communication
- Memiliki pengetahuan yang cukup baik dalam menerapkan best practice dari fitur fitur Node js
- Mampu proaktif mempelajari fitur fitur terbaru /terupdate dari Node.js
- Backend (GO)
 - Mengetahui sebagian besar seluk- beluk GO, dan bagaimana penggunaannya
 - Mampu menerapkan fitur Go Advanced seperti Go Modules, Marshaling & Unmarshalling JSON, Goroutines, Type, Type Assertions, buffer, Channels dan lain lain
 - Mampu menerapkan Build CLI dan ORM
 - Mampu menerapkan web framework seperti Gorilla, Buffalo,
 - Mampu menerapkan Logging seperti zep
 - Mampu menerapkan Real time communication seperti Melody, Centrifugo
 - Memiliki pengetahuan yang cukup baik dalam menerapkan best practice dari fitur fitur Go
 - Mampu proaktif mempelajari fitur fitur terbaru /terupdate dari GO

Senior Developer

- Frontend (React)
 - Memiliki pengetahuan yang tinggi tentang bagaimana kerangka kerja react dapat dimaksimalkan secara efektif dan teradaptasi dengan baik pada berbagai jenis produk berdasarkan requirement
 - Mampu memiliki pengetahuan dan menerapkan salah satu bahasa pemrograman/framework selain bahasa utama yang dipakai seperti fitur fitur dasar Vue. js ataupun yang lain
 - Mampu membuat pilihan-pilihan secara pragmatis dalam menggunakan best practices
- Frontend (Vue)
 - Memiliki pengetahuan yang tinggi tentang bagaimana kerangka kerja Vue dapat dimaksimalkan secara efektif dan teradaptasi dengan baik pada berbagai jenis produk berdasarkan requirement
 - Mampu memiliki pengetahuan dan menerapkan salah satu bahasa pemrograman/framework selain bahasa utama yang dipakai seperti fitur fitur dasar React. js atau yang lain.
 - Mampu membuat pilihan-pilihan secara pragmatis dalam menggunakan best practices
- Mobile (React native)
 - Memiliki pengetahuan yang tinggi tentang bagaimana kerangka kerja React Native dapat dimaksimalkan secara efektif dan teradaptasi dengan baik pada berbagai jenis produk berdasarkan requirement
 - Mampu memiliki pengetahuan dan menerapkan salah satu bahasa pemrograman/framework selain bahasa utama yang dipakai seperti fitur fitur dasar Flutter / Kotlin atau yang lain.
 - Mampu membuat pilihan-pilihan secara pragmatis dalam menggunakan best practices
- Mobile (Flutter)
 - Memiliki pengetahuan yang tinggi tentang bagaimana kerangka kerja Flutter dapat dimaksimalkan secara efektif dan teradaptasi dengan baik pada berbagai jenis produk berdasarkan requirement
 - Mampu memiliki pengetahuan dan menerapkan salah satu bahasa pemrograman/framework selain bahasa utama yang dipakai seperti fitur fitur dasar React native / Kotlin atau yang lain

- Mampu membuat pilihan-pilihan secara pragmatis dalam menggunakan best practices
- Backend (Node.js)
 - Memiliki pengetahuan yang tinggi tentang bagaimana kerangka kerja Node.js dapat dimaksimalkan secara efektif dan teradaptasi dengan baik pada berbagai jenis produk berdasarkan requirement
 - Mampu memiliki pengetahuan dan menerapkan salah bahasa pemrograman/framework selain bahasa utama yang dipakai seperti fitur fitur dasar Flutter / Vue. js / GO / React js / React native atau yang lain
 - Mampu membuat pilihan-pilihan secara pragmatis dalam menggunakan best practices
- Backend (GO)
 - Memiliki pengetahuan yang tinggi tentang bagaimana kerangka kerja Node.js dapat dimaksimalkan secara efektif dan teradaptasi dengan baik pada berbagai jenis produk berdasarkan requirement
 - Mampu memiliki pengetahuan dan menerapkan salah bahasa pemrograman/framework selain bahasa utama yang dipakai seperti fitur fitur dasar Flutter / Vue. js / Node. js / React js atau yang lain
 - Mampu membuat pilihan-pilihan secara pragmatis dalam menggunakan best practices

TECHNICAL QUALITY

Definisi :

Kemampuan dalam memastikan kode memiliki kualitas yang baik, efisien, dan mempertimbangkan maintainable, readable, reusable dan testable

Junior Developer

- Mampu mengikuti coding/development standard dari yang sudah ada
- Mampu mempertimbangkan penulisan struktur kode yang readable seperti fungsi dikelompokkan sesuai dengan penggunaan atau logic dan lain lain
- Mampu memodifikasi komponen komponen yang sudah ada.

Medior Developer

- Mampu konsisten menerapkan penulisan fungsi yang mudah dibaca, mudah diuji, tidak panjang dan menggunakan bahasa bahasa yang mudah dimengerti developer lain.
- Mampu secara konsisten menuliskan unit testing setiap fungsional yang dibuat dan muncul sebagai good unit test cases untuk code sendiri yang ditulis
- Mampu mendemonstrasikan dan mengaplikasikan pengetahuan dasar mengenai pendekatan Test-Driven

Senior Developer 1

- Mampu konsisten memprioritaskan kode yang reusable dan maintainable saat membuat dan menentukan komponen
- Mampu menuliskan fungsi yang terorganisir menjadi berbagai bagian dan well commented dengan petunjuk dari berbagai sumber file

- Mampu membuat code yang minim circular dependencies seperti library, build, third-party code ditempatkan kedalam folder yang sesuai
- Mampu menuliskan kode yang tidak mempunyai deep nesting atau loncat
- Mampu menuliskan unit test sesuai dengan TDD

Senior Developer 2

- Mampu menguasai secara mendalam dalam menghitung dan analisis kompleksitas $O(n)$ dari algoritma yang digunakan
- Mampu membantu team atau klien menggunakan pendekatan unit test seperti TDD untuk memastikan kualitas development yang baik.
- Mampu memandu anggota team dalam teknis penulisan unit test secara efektif,
- Mampu menetapkan pendekatan berbasis component-driven dan mengimplementasikan di beberapa project

TOOLING

Definisi :

Kemampuan menggunakan version control system dan tools untuk mendukung development seperti GIT, Webpack, package management

Junior Developer

- Mampu mengoperasikan di local environment dengan dukungan dari tim, seperti melakukan konfigurasi atau enabling/disabling debug mode
- Mampu menerapkan operasi GIT dasar seperti, cloning, fetching code, branching, committing, pushing, pulling dan merging
- Mampu memahami fundamental dari package management dan menggunakannya dengan tools terkait untuk pengoperasiannya
- Mampu menggunakan build tools seperti npm script, Webpack, esbuild, atau Linters & Formaters (Frontend)

- Familiar dengan project tools umum yang digunakan seperti IDE
- Mampu menggunakan CLIs

Medior Developer

- Mampu mengkloning project dan bekerja dengan beberapa macam environment dengan minimal bantuan dari developer lain
- Mampu menerapkan fundamental rebasing seperti memahami struktur commit, squashing dan tahu perbedaan rebasing dan merging
- Mampu menerapkan strategi branching di berbagai project/flow requirement yang berbeda
- Mampu menggunakan package management strategi di berbagai project/flow requirement yang berbeda
- Mampu menambahkan dependensi baru ke package management dalam penggunaannya, dan menyelesaikan konflik dependensi tersebut
- Mampu mengetahui dan menggunakan project tools secara efektif, misalnya dapat menggunakan IDE shortcut untuk mempercepat kecepatan pemrograman
- Mampu mengetahui cara pengoperasian CI/CD
- Mampu menjalankan perintah CLI biasa, seperti perintah environment management, perintah file handling and perintah start/stop

Senior Developer

- Mampu mengetahui dan menggunakan sebagian besar fitur penting pada project tools, seperti memindahkan dan mengganti nama classes pada IDE untuk re faktor, memeriksa log build untuk menemukan masalah pada pipeline
- Mampu menyusun kode menjadi sensible dan logical atomic yang masuk akal dan logis sebagai hal yang biasa
- Mampu membantu tim dalam mengoperasikan GIT,
- Mampu membantu tim dalam penggunaan package management,
- Mampu menggunakan cli sebagai bagian dari pekerjaan sehari hari dan membantu tim dalam penggunaan build tool atau CLIs,

- Mampu mengaplikasikan advance git seperti cherry-picking , penggunaan reflog ataupun rebasing

PERFORMANCE

Definisi :

Kemampuan dalam mendiagnosis atau perbaikan performance issue terhadap kode aplikasi yang baru atau yang sudah ada sebelumnya dan penggunaan tools yang sesuai untuk meng asses performance aplikasinya saat proses development.

Junior Developer

- Mampu mempelajari bagaimana proses implementasi untuk meningkatkan performance aplikasi dengan banyak bantuan oleh senior developer

Medior Developer

- Mampu konsisten menulis kode yang efisien, menghindari masalah performance issue yang biasa terjadi
- Mampu aware terhadap caching tetapi tidak selalu dapat memilih pendekatan terbaik
- Mampu menggunakan profiling tools di level dasar
- Mampu mengetahui data data vital web dan menguasai topik/konteks pada produk domain yang mungkin mempengaruhi performance issue

Senior Developer

- Mampu mengidentifikasi dan memilih berbagai macam solusi terhadap masalah performance yang biasa terjadi
- Mampu membantu klien untuk memprioritaskan perbaikan kode untuk mengoptimalkan performance
- Mampu memberi edukasi terhadap klien seberapa penting dan pengaruhnya terhadap performance
- Mampu mendukung junior developers untuk belajar meningkatkan skil mereka yang berkaitan dengan performance aplikasi

- Mampu memanfaatkan berbagai macam performance analysis tools

SECURITY

Definisi :

Kemampuan dalam memahami implikasi fundamental keamanan, serangan-serangan yang biasa terjadi dan cara menangani data sensitif dengan hati-hati

Medior Developer

- Mampu mengetahui pengetahuan dasar untuk menghindari kerentanan saat mengembangkan aplikasi
- Mampu mengetahui cara mengakses/mengirim data yang sifatnya sensitif dengan aman

Senior Developer

- Mampu mengetahui tentang vektor serangan serangan yang biasa terjadi dan cara mengatasinya seperti XSS, CSRF, perlindungan login, serialisasi objek tidak aman dan lain lain
- Mampu mengatasi masalah yang diangkat oleh audit keamanan
- Mampu menggunakan tools dan pendekatan untuk memastikan aplikasi dan server aman dari eksploitasi
- Mampu mengikuti security checklist dan tahu bagaimana menerapkan semua poin checklist (jika ada)

FRONTEND & MOBILE FOCUSED

Definisi :

Kemampuan dalam menginterpretasikan dan implementasi desain secara akurat (Slicing) dan integrasi API.

Junior Developer

- Mampu menerapkan konversi dari desain ke HTML dengan baik
- Mampu menerapkan HTML best practice HTML dan Forms & validations,
- Mampu menggunakan CSS framework
- Mampu melakukan styling seperti floats, positioning, display, Box Model, Grid dll
- Mampu menginterpretasikan dan mengikuti panduan desain visual dan branding secara konsisten
- Mampu memahami bagaimana penerapan responsive design untuk berbagai browser dan perangkat
- Mampu melakukan integrasi API yang berisi data yang sederhana seperti fetching API sederhana, getlist dan lain lain

Medior Developer

- Mampu menciptakan representasi desain secara akurat warna, positioning, pixel perfect dan lain lain
- Mampu menerapkan fitur yang kompatibel dengan berbagai browser dan perangkat, dan dapat menguji kompatibilitasnya
- Mampu mengimplementasikan visual desain yang cukup kompleks
- Mampu menginterpretasikan fitur fitur yang tidak terdokumentasikan dari desain / flow requirement
- Mampu memilih penggunaan CSS framework untuk styling secara efektif sesuai dengan kebutuhan user/ flow requirement
- Mampu memberikan rekomendasi pada desain untuk membantu meningkatkan efisiensi, kegunaan dan aksesibilitas
- Mampu melakukan integrasi API yang berisi banyak data data yang cukup kompleks

- Mampu melakukan integrasi ke platform third party

Senior Developer

- Mampu memahami bagaimana fitur berkaitan dengan kebutuhan pengguna atau klien dalam konteks yang lebih luas, sehingga mampu membuat keputusan yang mempertimbangkan pengguna secara konsisten
- Mampu implementasi visual desain dan animasi yang kompleks
- Mampu memberi rekomendasi UI seperti microinteraction atau animasi untuk meningkatkan user experience
- Mampu menerapkan penggunaan multi-theme system
- Mampu menerapkan pendekatan desain enhancement & degradation secara indah, dan memahami mana yang akan digunakan berdasarkan kasus kasus yang rumit/unik
- Mampu menerjemahkan feedback dari klien dan terimplementasi oleh rekomendasi dengan berbagai alternatif desain.
- Mampu melakukan integrasi API yang berisi banyak banyak data yang kompleks

BACKEND TECHNICAL FOCUSED

Definisi :

Kemampuan dalam implementasi pembuatan API, penguasaan dengan data storage system, penguasaan metode interaksi antar penyimpanan data serta bagaimana implementasi sistem arsitektur.

Junior Developer

- Mampu mengimplementasikan pembuatan API
- Mampu melakukan operasi penyimpanan dan pengambilan data
- Mampu memahami bagaimana struktur database bekerja
- Mampu menggunakan Relational Databases, NoSQL Database

Medior Developer

- Mampu merancang struktur sederhana yang sesuai dengan domain aplikasi dalam penyimpanan data dan memahami prinsip prinsip inti untuk membuatnya efisien
- Mampu menggunakan tools dan library yang biasa digunakan untuk berinteraksi dengan penyimpanan data,
- Mampu memeriksa konten penyimpanan data untuk menghubungkan behavior aplikasi dengan data yang disimpan
- Mampu memilih tempat yang tepat untuk menyimpan data, memahami implikasinya misalnya seperti kapan harus menggunakan cookie, local storage atau session
- Mampu mengetahui cara menyimpan atau mereferensikan data sensitif dengan tepat dan aman.
- Mampu integrasi dengan Third party dan cara penerapannya

Senior Developer

- Mampu merencanakan kode untuk meminimalkan dampak penggunaan penyimpanan data, seperti mengambil data dalam jumlah yang besar sebagai kumpulan data dibanding mengambil satu persatu.
- Menggunakan teknik untuk meningkatkan kinerja operasi penyimpanan data, seperti penggunaan struktur tabel yang dinormalisasi pada relational database yang tepat
- Mampu menggunakan pendekatan untuk menganalisis interaksi penyimpanan data untuk mengidentifikasi bottleneck dan peluang untuk optimasi
- Mampu menyadari manfaat dan kerugian dari platform atau library berinteraksi dengan penyimpanan data. misalnya pelacakan transaksi database, atau bagaimana skema data dapat dibangun.
- Mengetahui teknik cara memantau dan mendiagnosis masalah seperti penggunaan MySQL slow query log, atau browser developers tools untuk memeriksa cookie dan local storage
- Mampu memahami tipe tipe design pattern seperti behavioral, creational, atau structural



- Mampu membantu Junior / Mid developer untuk mendesain sistem dengan baik