



Glossário de Desenvolvimento

Glossário de Desenvolvimento	1
Ferramentas de programação	3
Sistema operacional	4
Terminal	4
Editor de Texto	4
Compilador	4
Linguagem de máquina	4
Interpretador	4
Browser ou Navegador	4
Plugin	4
Conceitos de programação	5
Algoritmo	5
Linguagem	5
Código	5
Código Fonte	5
Código de Máquina	5
Código Binário	5
Bug	5
Refatoração	5
Frameworks	5
Queries	5
Front-End	5
Back-end	6
Full-stack	6
SSL (Secure Sockets Layer)	6
SOLID (Single-responsibility, Open-closed, Liskov substitution, Interface segregation e Dependency Inversion principles)	6
Biblioteca	6
Front-End	6
HTML (Linguagem de Marcação de Hipertexto)	6
CSS (Folhas de Estilo em Cascata)	6
JavaScript	6
UI Design (Design da Interface do Usuário)	6
UX Design (Design da Experiência do Usuário)	7
Back-End	7
Servidores	7



Banco de Dados	7
Banco de Dados Relacional (SQL)	7
Banco de Dados não Relacional (NoSQL)	7
SQL (Linguagem de Consulta Estruturada)	7
Sistema de Gerenciamento de Banco de Dados (SGBD)	7
API (Application Programming Interface)	7
Serviço Web	7
Webhook	8
HTTP (HyperText Transfer Protocol)	8
REST (REpresentational State Transfer)	8
Restful	8
CRUD (Create, Read, Update and Delete)	8
Programação	8
Habilidades	8
Mindsets e Metodologias	8
Práticas	8
Pair programming	9
Sprint	9
Kanban	9
Ágil	9
Scrum	9
MVP (Produto Mínimo Viável)	9
Papéis	9
Product Owner (PO)	9
Regras de Negócio	9
Scrum Master	9
Dev. Team	10
Cerimônias	10
Daily Meeting (DM)	10
Grooming / Refinamento	10
Planning	10
1:1	10
Artefatos	10
Product Backlog	10
Sprint Backlog	10
Definição de Pronto	10
Incremento	10
Git	10
Conceitos	11



Git	11
GitHub	11
Diretório	11
Branch	11
Ações no Git	11
Git fork	11
Git clone	11
Git merge	11
Git fetch	11
Git pull	11
Git add	12
Git commit	12
Git push	12
Git branch	12
Git checkout	12
Deploy de Aplicação e boas práticas de entrega	12
Deploy	12
QA (Quality Assurance)	13
Code Climate (CC)	13
Cognitive Complexity	13
Cyclomatic Complexity	13
Linter	13
Code Smell	13
Continuous Integration (CI)	13
Testes automatizados	13
Code Review (CR)	13
Test Driven Development (TDD)	14
Teste Unitário	14
Mock	14

Ferramentas de programação

As ferramentas de programação são softwares ou programas que as pessoas da área usam para criar, depurar, processar ou otimizar alguma etapa do processo de trabalho. Algumas das principais são:



Sistema operacional

– Sistema que faz comunicação entre o hardware, gerenciando e distribuindo seus recursos, e os demais softwares e cria uma plataforma comum a todos os programas utilizados. Exemplo: Windows 10, Ubuntu.

Terminal

– É uma Interface de Linha de Comando, isto é, uma interface (meio) para a pessoa enviar comandos ao computador.

Editor de Texto

– Um editor de texto é um aplicativo de edição de arquivos de texto. Isso inclui códigos de todos os tipos, que também são texto escrito num formato que o computador é capaz de entender.

Compilador

– Transforma o código de uma linguagem de programação para uma linguagem de máquina (ou alguma de nível mais baixo, ou seja, mais próxima do código binário que o hardware processa) que é lida pelo processador.

Linguagem de máquina

– Linguagem de mais baixo nível de entendimento para o ser humano, e a única, entendida pelo processador (CPU).

Interpretador

– Programas de computador que lêem um código fonte de uma linguagem de programação interpretada, ou seja, convertida em código binário uma linha por vez, ao invés de grandes blocos de linhas de cada vez, e o converte em código executável.

Browser ou Navegador

– Programa desenvolvido para permitir a navegação pela web e visualização de suas páginas.

Plugin

– Uma extensão de código que pode ser adicionada a um código em que você está trabalhando. O plugin é uma extensão intrínseca ao contexto, por exemplo um plugin de um navegador: Ele não funcionará em outro navegador. Além disso, ele funciona consumindo código ou APIs da aplicação principal.



Conceitos de programação

Algoritmo

- É basicamente um conjunto de passos para certa tarefa.

Linguagem

- Conjunto de definições (sintáticas e semânticas) usadas para fazer um texto que o computador é capaz de entender, ou seja, um programa de computador.

Código

- Conjunto de palavras ou símbolos contendo instruções para o computador.

Código Fonte

- Código de um programa. O computador o usa para gerar uma versão do programa que é capaz de processar.

Código de Máquina

- Código que a máquina consegue entender e executar.

Código Binário

- Código feito por apenas 0 e 1 que tem como função codificar algo. É a base de toda a lógica do computador. Tudo vira, no fim das contas, código binário.

Bug

- Problema no código que faz com que ele não execute sua função corretamente.

Refatoração

- Processo de alterar um programa para melhorar sua estrutura interna sem alterar o seu funcionamento. Essas melhorias podem ser de performance ou para facilitar trabalhos futuros neste código, por exemplo.

Frameworks

- São como coleções de ferramentas, componentes e soluções que você pode encontrar em um mesmo pacote (o framework) que facilitam a criação de aplicações complexas.

Queries

- Pedido de informação e/ou dados para banco de dados.

Front-End

- É a parte de uma aplicação web que a pessoa que utiliza, vê e interage. É executada pelo *browser*.



Back-end

– O que a pessoa que usa não vê num software, como servidores e banco de dados. É o processamento “por trás dos panos”.

Full-stack

– Junção de Front-End e Back-End.

SSL (Secure Sockets Layer)

– É um tipo de segurança padrão para garantir que todos os dados sejam criptografados no trânsito entre um site e um navegador por exemplo.

SOLID (Single-responsibility, Open-closed, Liskov substitution, Interface segregation e Dependency Inversion principles)

– Esses princípios, quando combinados, são tidos como boas práticas para facilitar a quem programa desenvolver softwares que são mais fáceis de se manter e expandir.

Biblioteca

– Pedacos de códigos (como funções, por exemplo) já programados e que podem ser usados em outros códigos. A biblioteca é uma extensão de fora do contexto, ou seja, ela pode ser usada e desplugada de um contexto para outro. As bibliotecas são consumidas pela aplicação principal.

Front-End

HTML (Linguagem de Marcação de Hipertexto)

– Uma linguagem de marcação que é utilizada como estrutura do Front-End de um site.

CSS (Folhas de Estilo em Cascata)

– É uma linguagem de estilo que define a aparência de documento web HTML. Exemplo: manipulação da posição dos elementos, cores, fontes, fundos, etc.

JavaScript

– É uma linguagem que viabiliza a criação de páginas interativas, ou seja, que são modificáveis de acordo com a interação de quem usa e que conectam o Front-End ao Back-End.

UI Design (Design da Interface do Usuário)

– É o desenho e execução de uma interface para o usuário. Exemplo: Como garantir que um botão vai ser visto e vai ser apertado.



UX Design (Design da Experiência do Usuário)

– É o desenho e execução de uma experiência para o usuário. Como por exemplo, desenhar as ações que farão com ele se sintam bem ao entrar numa página, ou engajado, etc.

Back-End

Servidores

– Máquina onde roda o Back-End de um site, por exemplo. Ou um Back-End acessível a outras aplicações diversas.

Banco de Dados

– Agrupamento (em formato de tabelas, documentos, colunas e etc...) de informações que se relacionam e tratam de um mesmo assunto.

Banco de Dados Relacional (SQL)

– Banco de dados onde os mesmos são guardados em tabelas que possuem “relações” entre si. Por exemplo, uma tabela de produtos e de compras. A tabela de compras pode ter vários produtos e um produto pode estar relacionado a várias tabelas.

Banco de Dados não Relacional (NoSQL)

– Quando o banco de dados não guarda os dados em tabelas, e sim em chave-valor, colunas, etc.

SQL (Linguagem de Consulta Estruturada)

– Linguagem de banco de dados relacional.

Sistema de Gerenciamento de Banco de Dados (SGBD)

– Responsável por gerenciar o acesso, a persistência, a manipulação e a organização dos dados.

API (Application Programming Interface)

– Interface que conecta dois programas, realizando tanto a comunicação entre eles e especificando quanto definindo como devem interagir.

Serviço Web

– Solução utilizada na integração de sistemas e na comunicação entre aplicações diferentes (é um programa web de Back-End, um tipo de API que necessariamente precisa de uma rede para se comunicar).



Webhook

– É uma forma de receber informações quando um evento (enviar um email, por exemplo) acontece. Por isso o webhook é, na prática, uma forma de transmissão de informações entre dois sistemas passiva, pulando o passo de request (requerimento) que uma API faz. Basicamente: um sistema disponibiliza um canal para receber informações (o webhook) e outros enviam para esse sistema informações através dele.

HTTP (HyperText Transfer Protocol)

– Protocolo que especifica como clientes e servidores se comunicam entre si.

REST (REpresentational State Transfer)

– Estilo de arquitetura de serviços web que padroniza algumas características tidas como boas práticas.

Restful

– É uma aplicação web que realiza a implementação do padrão REST.

CRUD (Create, Read, Update and Delete)

– Quatro operações básicas de interação com uma base de dados (criação, consulta, atualização e destruição de dados).

Programação

Habilidades

Hardskills

– Habilidades Técnicas (por exemplo, programar em JavaScript).

Softskills

– Competências comportamentais (por exemplo, saber criticar e receber críticas, saber se explicar, saber criar boas relações com as pessoas).

Mindsets e Metodologias

Práticas



Pair programming

- Técnica do desenvolvimento ágil de software onde duas pessoas utilizam um mesmo computador, para programar alguma tarefa.

Sprint

- Ciclo de tempo dentro do qual se realiza um conjunto de atividades.

Kanban

- Sistema simples e visual de organizar as tarefas e o fluxo de trabalho.

Ágil

- Conjunto de métodos e princípios que consiste na ideia de desenvolvimento iterativo (isto é, onde requisitos e soluções são identificados e cumpridos de maneira cíclica) a partir da colaboração de times auto-gerenciáveis e multifuncionais.

Scrum

- Framework para gestão e planejamento de produtos e projetos.

MVP (Produto Mínimo Viável)

- Maneira de se testar modelos de negócios e conceitos diretamente com o cliente, a partir do lançamento de uma versão mínima viável do produto. Exemplo: Em vez de fazer um site como o que a Airbnb, os fundadores fizeram um site simples com poucas funcionalidades.

Papéis

Product Owner (PO)

- Pessoa responsável por definir quais as prioridades do product backlog e registrar de todos os requisitos, regras de negócio e prioridades das entregas de um produto.

Regras de Negócio

- Requisitos de como uma determinada funcionalidade ou produto deveria se comportar dado um determinado critério.

Scrum Master

- É a pessoa responsável por garantir que todos estejam aderentes ao processo de desenvolvimento, que sigam suas cerimônias e que garantam a organização e realização dos seus artefatos.



Dev. Team

- Time de desenvolvimento (pessoas que colaboram com o projeto).

Cerimônias

As cerimônias são os rituais, reuniões e eventos que acontecem em um local de trabalho, estudo ou convivência de modo geral. Geralmente, empresas mais tecnológicas têm bem estruturadas quais são suas cerimônias e o porquê de cada uma delas. Alguns exemplos são:

Daily Meeting (DM)

- Reunião diária de dez a quinze minutos para saber o que se fez no dia anterior, o que se vai fazer no dia seguinte e o que é impedimento seu para fazer isso.

Grooming / Refinamento

- Reuniões sobre como melhorar o Product [Backlog](#).

Planning

- Reunião onde se descreve as funcionalidades de maior prioridade (e que devem ser feitas) para a equipe para o próximo sprint.

1:1

- Reunião periódica entre uma pessoa que gere o time e uma pessoa liderada.

Artefatos

Product Backlog

- O que está para ser feito no produto.

Sprint Backlog

- O que está para ser feito na sprint.

Definição de Pronto

- O como que o time define o que foi feito.

Incremento

- O que foi feito e gerou valor ao usuário.



Git

Conceitos

Git

– Sistema de Controle de Versão (VCS) distribuído, ou seja, de uso compartilhado por várias pessoas de uma equipe, que faz o acompanhamento de mudanças em arquivos e pastas de uma aplicação.

GitHub

– Plataforma de hospedagem de código-fonte de uma aplicação com controle de versão usando o Git.

Diretório

– Um tipo de pasta do computador que pode ter dentro de si outros diretórios e arquivos.

Branch

– É uma ramificação do código do controle de versão, ou seja, uma versão alterada do código “principal” salvo lá.

Índice – Funciona como uma área do git no qual os arquivos possam ser enviados ao GitHub. É como se fosse uma sala de embarque.

Ações no Git

Git fork

– Comando que permite “copiar” um projeto em código aberto para si (cria um novo repositório que é uma cópia do projeto “forkado”).

Git clone

– Comando que permite copiar um repositório remoto da nuvem e colar na sua máquina local.

Git merge

– Comando usado para colocar as mudanças de uma branch em outra.

Git fetch

– Comando para pegar alterações dos seus projetos remotos.



Git pull

– Incorpora as alterações adicionadas a um repositório remoto na branch atual. git pull = git merge + git fetch.

Git add

– Comando usado para passar a monitorar um novo arquivo no índice para um novo commit, ou adicionar ao índice as alterações mais recentes feitas a um arquivo.

Git commit

– Armazena o conteúdo atual do índice em um pacote de mudanças, juntamente com uma mensagem de registro da pessoa que as descreve.

Git push

– Comando em que você transfere commits do seu repositório local para um repositório remoto.

Git branch

– Comando para listar as branches locais.

Git branch nome-da-branch-nova – Serve para criar uma nova branch.

Git checkout

– Você pode mudar de branch com esse comando ou limpar as alterações de arquivos mais recentes ainda não adicionadas ao índice.

Pull Request

– É um pedido que você faz para a administradora da branch que você quer modificar adicionar (puxar para si) a ela as alterações que você fez.

Merge Pull Request

– Inserir as alterações do código de uma Pull Request em sua branch “alvo”.

Deploy de Aplicação e boas práticas de entrega

Deploy

– O deploy consiste no processo que coloca uma aplicação no ar, tornando-a acessível a quem usa.



QA (Quality Assurance)

- Conjunto de atividades exercidas para que o produto ou serviço oferecidos sejam de qualidade.

Code Climate (CC)

- Programa que lê um código e analisa a forma como foi escrito em busca de pontos de melhoria. Ele faz isso analisando o seu código fonte contra algumas regras de código e estilo que verificam como o código foi escrito.

Cognitive Complexity

- É uma medida de quão difícil é para uma pessoa entender o código.

Cyclomatic Complexity

- Métrica que mede a quantidade de caminhos independentes pelo código e assim, também, medindo o quão difícil testá-lo.

Linter

- Ferramenta que analisa o código fonte em busca de erros, problemas de estilo e construções tidas como problemáticas do ponto de vista de desenvolvimento.

Code Smell

- Característica de um código que o faz suspeito de possuir ou se tornar causador de problemas no futuro.

Continuous Integration (CI)

- Prática de colocar em produção, de forma automática, as novas modificações feitas pelos colaboradores do projeto de forma contínua (em geral, uma vez por dia no mínimo), ou seja, continuamente e sem a necessidade de períodos fora do ar.

Testes automatizados

- O teste automatizado de software é um programa que executa testes pré-programados em um aplicativo de software antes de serem liberados para deploy.

Code Review (CR)

- A Code Review (revisão de código) é uma prática da [engenharia de software](#) que ajuda a garantir a qualidade do código. Essa prática consiste basicamente em, um ou mais desenvolvedores podem comentar no código de outro.



Test Driven Development (TDD)

– Desenvolvimento Orientado por Testes. Técnica de desenvolvimento em que a produção e o deploy do software é baseada em testes.

Teste Unitário

– Testes isolados entre si que são feitos com pequenas unidades de códigos para validar seu funcionamento.

Mock

– Programa que serve para simular outros códigos. São muito usados para garantir isolamento de testes unitários.

Referências:

<https://blog.betrybe.com/carreira/glossario-de-tecnologia/>