



# Introdução ao Desenvolvimento Moderno de Software

Denilson Bonatti  
Tech Lead - DIO

# DIO Bank

Bancos digitais são instituições financeiras que executam suas atividades de forma 100% online. Ou seja, praticamente tudo que o cliente precisa é feito por um celular via aplicativo ou no computador via Browser.



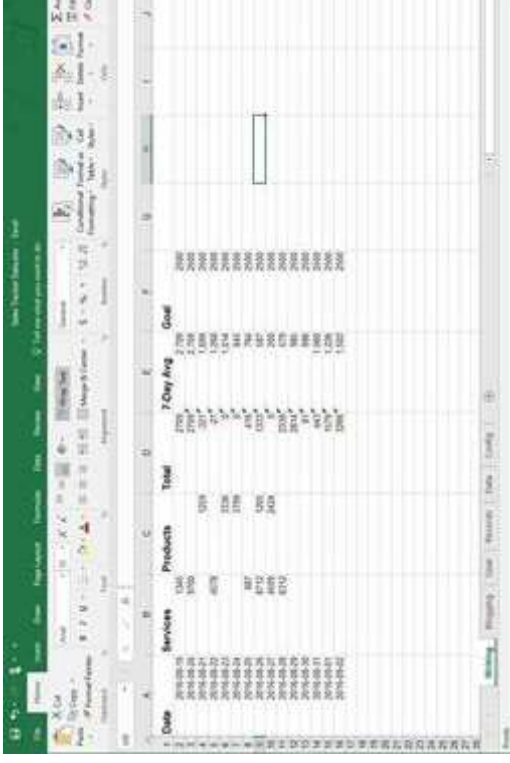


# Sistema Desktop X Sistema Web

Ao iniciar o desenvolvimento de um software, o primeiro passo é definir a(s) plataforma(s) onde este software será executado.

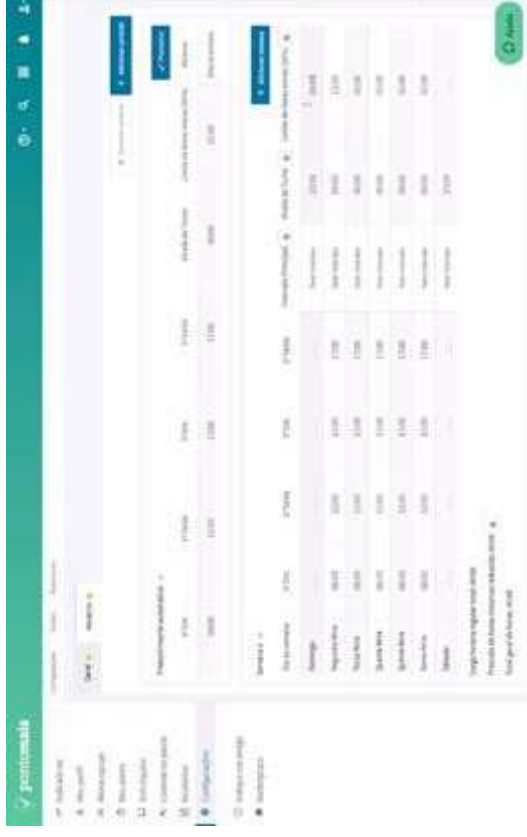
# Sistema Desktop

Sistemas desktop são sistemas autônomos que podem ser instalados no computador. Esta instalação normalmente é realizada por um arquivo executável. Como exemplo, temos o Microsoft Word, Microsoft Excel, Anti-vírus e jogos.



# Sistema Web

Sistemas baseados em tecnologia web, podendo ser utilizados remotamente através de qualquer navegador de internet, sem a necessidade de instalação e atualização local.



The screenshot displays a web-based financial management system. The interface includes a sidebar menu on the left with options like 'Início', 'Relatório', 'Configuração', and 'Ajuda'. The main area shows a 'Resumo de Contas' (Summary of Accounts) table with columns for 'Conta', 'Saldo', 'Débito', and 'Crédito'. Below this, there are several smaller tables and charts, including a 'Gráfico de Contas' (Accounts Chart) and a 'Resumo de Contas' (Summary of Accounts) table. The interface is clean and professional, with a green header bar and a white background.

Conta	Saldo	Débito	Crédito
Conta 1	1000	1000	1000
Conta 2	2000	2000	2000
Conta 3	3000	3000	3000
Conta 4	4000	4000	4000
Conta 5	5000	5000	5000
Conta 6	6000	6000	6000
Conta 7	7000	7000	7000
Conta 8	8000	8000	8000
Conta 9	9000	9000	9000
Conta 10	10000	10000	10000

# E as aplicações móveis?

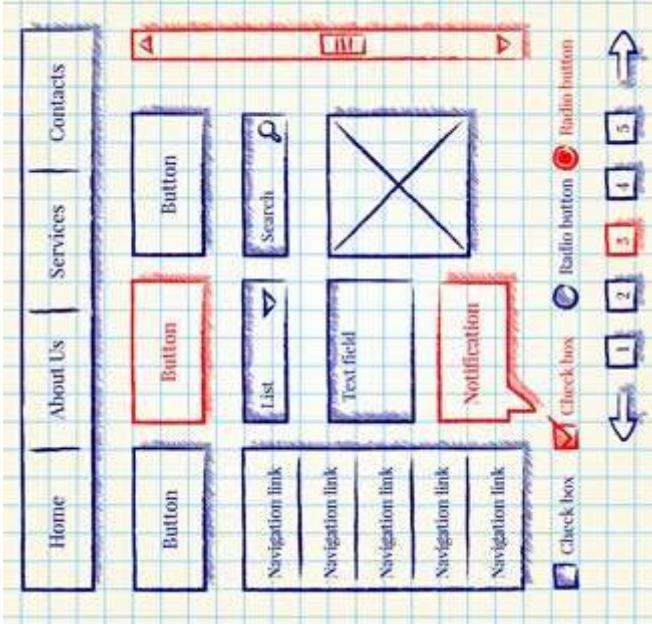
Uma aplicação móvel ou aplicativo mobile é um software desenvolvido para ser instalado em smartphones e/ou tablets. É baixado de uma loja on-line, como Google Play ou App Store, direto para o seu dispositivo portátil.



# UX Design (User Experience)

O design da experiência do usuário (ou UX design) é o processo que visa melhorar a satisfação do usuário com um produto ou serviço, melhorando a usabilidade, a acessibilidade e até mesmo a satisfação proporcionada na interação.







COMPORTAMENTO

# UX

PESQUISA

PROTÓTIPOS

PERSONAS

OBJETIVOS

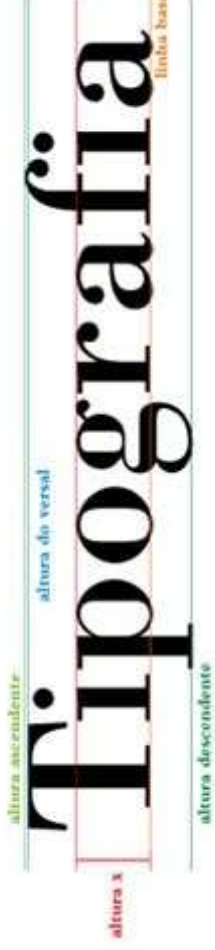
# UI Design (User Interface)

O designer de interface do usuário é o profissional responsável em criar o que o usuário irá ver e utilizar o produto. Este profissional entende padrões visuais que podem ajudar o usuário na experiência de utilização do software.

**Profissional focado em cores, tipografia, microinterações e estilos.**

# UI Design (User Interface)

UI designer, ou designer de interface do usuário, promove a criação e o desenvolvimento da interface explorada pelo usuário em um produto ou serviço.



COMPORTAMENTO

# UX

PESQUISA

PROTÓTIPOS

PERSONAS

OBJETIVOS



APARÊNCIA

# UI

DESIGN

TIPOGRAFIA

CORES

LAYOUTS

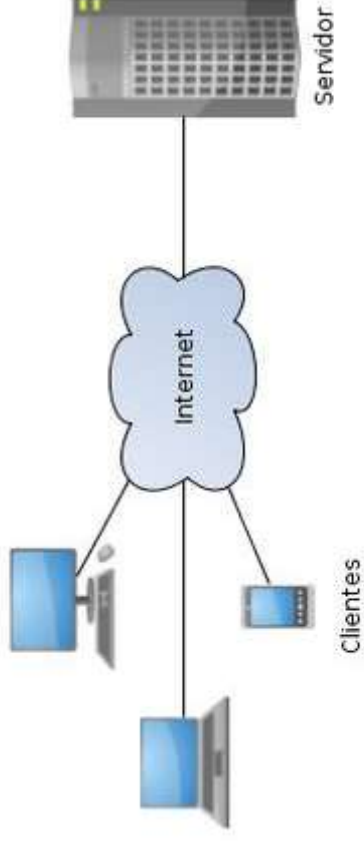
# Ferramentas utilizadas

1. Invision
2. Visio
3. Adobe XD
4. Figma



# Modelo Cliente-Servidor

O modelo cliente-servidor é uma estrutura de aplicação que distribui as tarefas e cargas de trabalho entre os fornecedores de um recurso ou serviço, designados como servidores, e os requerentes dos serviços, designados como clientes.



# Desenvolvedor Front End

O desenvolvedor Front End é que programa a parte visual de um site ou aplicativo, ou seja, aquilo que conseguimos interagir. Quem trabalha com Front End é responsável por desenvolver por meio de código uma interface gráfica, normalmente com as tecnologias base da Web (HTML, CSS e JavaScript).

# Onde que eu crio os códigos?



## Em um IDE!

Um ambiente de desenvolvimento integrado (IDE) é um software para criar aplicações que combina ferramentas comuns de desenvolvimento em uma única interface gráfica do usuário (GUI).



# Mas o que são os frameworks?

**Framework é, de forma básica, um facilitador.**

Ele traz diversas soluções já pré-definidas, que descomplicam o trabalho dos profissionais no desenvolvimento de aplicativos e outros projetos digitais.

Afinal, a atuação de um programador pode ter muito de criatividade, mas também traz aspectos mecânicos, de repetição de tarefas, que seriam maçantes sem a possibilidade de automatização.

Exemplos: Angular, Laravel e Vue

# Desenvolvedor Back End

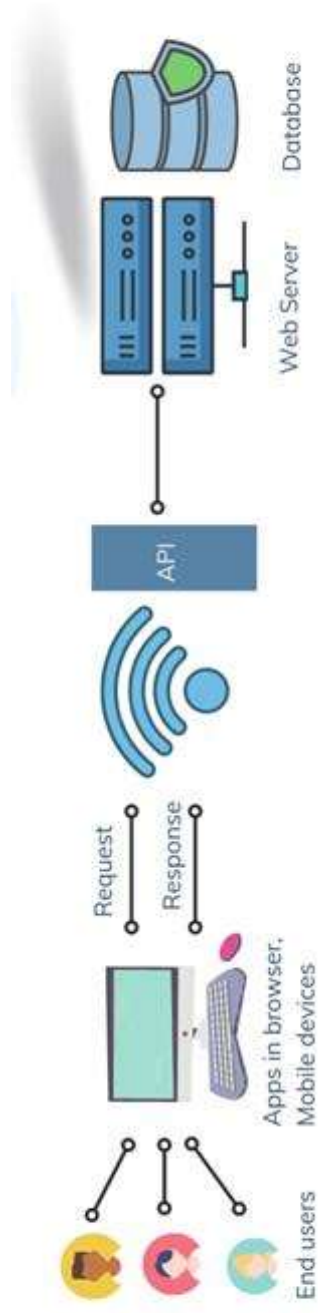
Back End, como o próprio nome sugere, vem da ideia do que tem por trás de uma aplicação. Pode ficar meio abstrato em um primeiro momento, mas pense que para conseguir usar o Facebook no dia a dia, os dados do seu perfil, amigos e publicações precisam estar salvos em algum lugar, sendo esse lugar um banco de dados e processados a partir de lá.

O Back End trabalha em boa parte dos casos fazendo a ponte entre os dados que vem do navegador rumo ao banco de dados e vice-versa, sempre aplicando as devidas regras de negócio, validações e garantias em um ambiente onde o usuário final não tenha acesso e possa manipular algo.

Exemplo: JAVA, PHP e C#

# O que é uma API?

Acrônimo de Application Programming Interface (interface de programação de aplicativos), um intermediário de software que permite que dois aplicativos conversem entre si. Cada vez que você usa um app como o Facebook, envia uma mensagem instantânea ou verifica a previsão do tempo em seu telefone, você está usando uma API.





# Desenvolvedor Full Stack

Um Desenvolvedor Full Stack é alguém que trabalha com o Back End do aplicativo, bem como o Front End. Desenvolvedores Full Stack precisam ter algumas habilidades em uma ampla variedade de linguagens de programação.

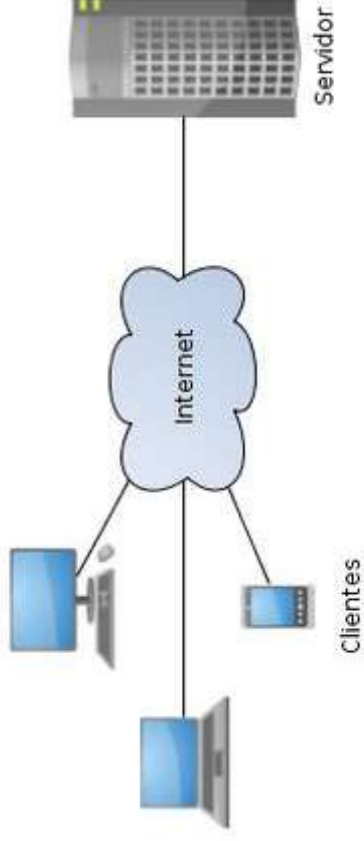
# Quality Assurance (QA)

QA pode ser definida como um conjunto de ações que as empresas realizam com o objetivo de entregar aos consumidores um produto ou serviço com alto nível de qualidade. No desenvolvimento de software, aplicar os métodos de QA geram confiança e segurança aos clientes, indicando que os seus produtos terão a qualidade esperada na etapa de implantação.

O profissional de Quality Assurance deve ter conhecimento sobre as atividades do projeto, além de ter um perfil analítico. Ele verifica se os padrões de qualidade estão sendo atendidos e se todos os requisitos mínimos esperados no produto serão entregues.

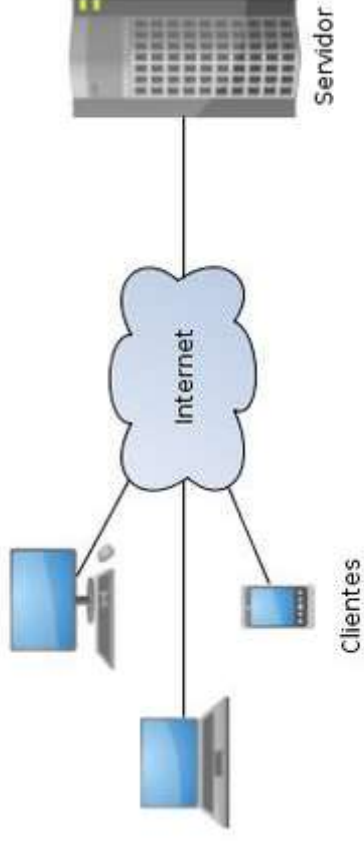
# Modelo Cliente-Servidor

O modelo cliente-servidor é uma estrutura de aplicação que distribui as tarefas e cargas de trabalho entre os fornecedores de um recurso ou serviço, designados como servidores, e os requerentes dos serviços, designados como clientes.



# Modelo Cliente-Servidor

O modelo cliente-servidor é uma estrutura de aplicação que distribui as tarefas e cargas de trabalho entre os fornecedores de um recurso ou serviço, designados como servidores, e os requerentes dos serviços, designados como clientes.









# Atividades e profissionais em nuvem privada

- 1 – Segurança da Tecnologia da Informação (lógica e física)
- 2 – Mão de obra especializada (software e hardware)
- 3 - Infraestrutura local

# Nuvem Pública

Os sistemas em nuvem são sistemas de armazenamento de dados disponibilizados via internet, em vez de servidores físicos tradicionais. Hoje, muitas organizações estão migrando o armazenamento de dados de servidores físicos para sistemas baseados em nuvem.





# Vantagens de migrar para a nuvem pública

- 1 – Preço (pague somente o que usar)
- 2 - Facilidade de contratação, configuração e infraestrutura
- 3 – Escalabilidade
- 4 - Performance

# Profissional de Cloud Computing

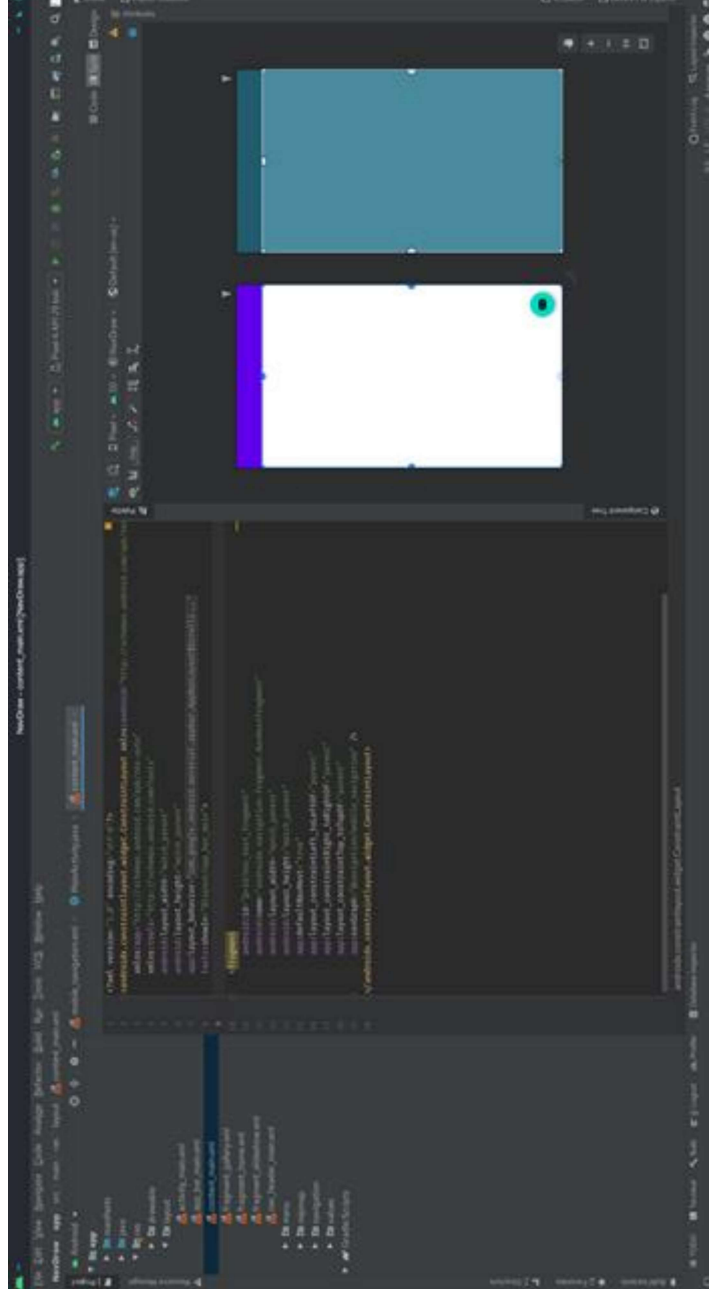


O profissional cloud computing é responsável pela infraestrutura de nuvem oferecida aos clientes. Mais do que desenhar sistemas ou ambientes de TI, ele escolhe as tecnologias que serão usadas, quais operadores são mais interessantes, como as peças vão ser integradas e, no fim, cuida do que foi construído.

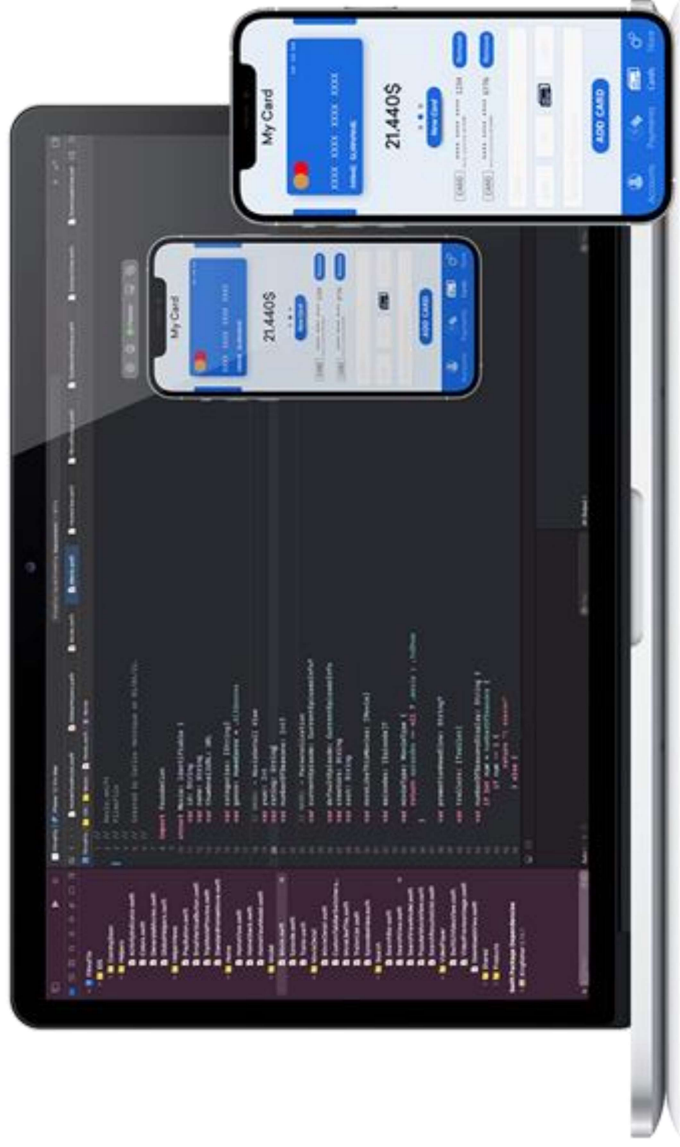
# Desenvolvimento mobile: Android



JAVA e/ou  
KOTLIN

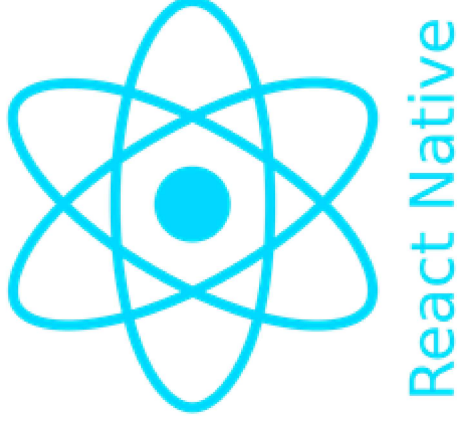


# Desenvolvimento mobile: IOS



SWIFT

# Desenvolvimento Híbrido (web)





# Conclusão

