

Aula 1

CONCEITOS E DEFINIÇÕES

A partir deste momento, você aprenderá o que é e como funciona um framework, além de quais são seus benefícios e seus desafios dentro do ambiente de programação.

46 minutos

INTRODUÇÃO

Os frameworks são considerados estruturas prontas cujo objetivo é facilitar o desenvolvimento de softwares que já foram criados anteriormente por outros programadores. Para não “reinventar a roda”, essas estruturas vêm para auxiliar e tornar mais fácil a criação de projetos, já que apresentam uma arquitetura pronta para ser utilizada.

A partir deste momento, você aprenderá o que é e como funciona um framework, além de quais são seus benefícios e seus desafios dentro do ambiente de programação. Para tanto, é necessário buscar muita leitura e utilizar o maior número de ferramentas possível para complementar os estudos, pois, dessa maneira, com certeza o aprendizado será certo. Uma dica é sempre dedicar tempo para pesquisar termos que não tenha visto, assim o conteúdo será melhor entendido. Bons estudos!

INTRODUÇÃO AO USO DE FRAMEWORKS: OBJETIVOS

Frameworks e sua utilização

A capacidade da computação e a largura de banda da rede aumentaram drasticamente na última década, mas o design e a implementação de softwares complexos continuam caros e sujeitos a erros. Grande parte do custo e do esforço empregados nessas áreas decorre da redescoberta e da reinvenção contínuas de conceitos e componentes essenciais em toda a indústria de software. Em particular, a crescente heterogeneidade de arquiteturas de hardware e diversidade de sistemas operacionais e plataformas de comunicação tornam difícil construir aplicativos corretos, portáteis, eficientes e baratos a partir do zero.

Para Fayad e Schmidt (1997), frameworks de aplicativos orientados a objetos são uma tecnologia promissora para reificar projetos e implementações comprovadas de software a fim de reduzir o custo e melhorar a qualidade dele. Um framework é uma estrutura de aplicativo reutilizável, "semicompleta", que pode ser especializada para produzir aplicativos personalizados.

Os principais benefícios de frameworks de aplicativos orientados a objetos derivam da modularidade, capacidade de reutilização, extensibilidade e inversão de controle que fornecem aos desenvolvedores. As estruturas aprimoram a modularidade, encapsulando detalhes de implementação voláteis por trás de interfaces estáveis. Além disso, a modularidade do framework ajuda a melhorar a qualidade do software ao localizar o impacto das mudanças de design e implementação, o que reduz o esforço necessário para entender e manter o software existente (FAYAD; SCHMIDT, 1997).

O início dos frameworks

Os primeiros exemplos do conceito de framework podem ser encontrados na literatura que tem origem no ambiente de programação *Smalltalk* e *Apple Inc*. O framework de interface do usuário *Smalltalk-80, Model-View-Controller (MVC)*, foi talvez o primeiro framework amplamente usado. O **MVC** foi usado para desenvolver interfaces de usuário, as quais foram divididas em três partes: modelos, visualizações e controladores. O modelo é um objeto de aplicativo independente da interface do usuário; a visualização gerencia uma região da tela e o controlador converte os eventos do usuário em operações em seu modelo e em sua visualização. A *Apple Inc*, por exemplo, desenvolveu um framework de interface do usuário do **Mac** que foi projetado para implementar aplicativos **Macintosh**. A partir disso, é possível afirmar que os frameworks não se limitam à estrutura da interface do usuário, mas existem também para inúmeras outras aplicações (MATTSSON, 1996).

Nesse sentido, as interfaces estáveis fornecidas por frameworks aprimoram a capacidade de reutilização, definindo componentes genéricos que podem ser reaplicados para criar novos aplicativos. A capacidade de reutilização da estrutura alavanca o conhecimento de domínio e o esforço anterior de desenvolvedores experientes para evitar repetição e revalidação de soluções comuns para requisitos recorrentes de aplicativos e desafios de design de software. A reutilização de componentes do framework pode gerar melhorias substanciais na produtividade do programador, bem como aprimorar a qualidade, o desempenho, a confiabilidade e a interoperabilidade do software.

Segundo Mattsson (1996), um framework aumenta a extensibilidade ao fornecer métodos de ganho explícitos, os quais permitem aos aplicativos estenderem suas interfaces estáveis. Os métodos sistematicamente desacoplam as interfaces estáveis e os comportamentos de um domínio de aplicativo das variações exigidas pelas instâncias de um aplicativo em um determinado contexto. A extensibilidade da estrutura é essencial para garantir a customização oportuna de novos serviços e recursos de aplicativos.

VIDEOAULA: INTRODUÇÃO AO USO DE FRAMEWORKS: OBJETIVOS

O vídeo deverá mostrar alguns conceitos iniciais de frameworks, e como eles podem contribuir para o início de projetos de software. Deverá ser abordado ainda quanto importante é a utilização de frameworks para o desenvolvimento de um software, tendo em vista que facilita a programação e diminui significativamente o tempo empregado nesse processo.

Vídeoaula: Introdução ao uso de frameworks: objetivos

Para visualizar o objeto, acesse seu material digital.

CONCEITOS DE FRAMEWORKS E A DIFERENÇA ENTRE FRAMEWORK, API E COMPONENTES

Conceitos de frameworks

Existem muitas definições, que ora diferem, ora se assemelham, quando se trata de frameworks. Para Firesmith (1994), por exemplo, framework é uma coleção de classes colaborativas, que capturam os padrões em pequena escala, e de mecanismos maiores, que implementam requisitos e projetos em comum. Já para Mattsson (1996), um framework é uma arquitetura (generativa) projetada para reutilização máxima, representada como um conjunto coletivo de classes abstratas e concretas, com potencial encapsulado e comportamento para especializações de subclasse.

Enquanto isso, para Fujioka (2015), um framework orientado a objetos é uma arquitetura que permite a reutilização de todo o sistema ou de parte dele. A representação é realizada por meio de um conjunto de classes abstratas e concretas que se relacionam da maneira como suas instâncias interagem a fim de fornecer uma solução reutilizável para um grupo de problemas inter-relacionados. Esse conjunto de classes deve ser extensível e flexível com o intuito de que algumas aplicações possam ser construídas sem muito esforço, pois grande parte da funcionalidade será reutilizada e não criada e terá apenas definidas as especificidades de cada software.

Conceitos de API

As interfaces de programação de aplicativos, ou APIs, simplificam o desenvolvimento e a inovação de softwares, permitindo que os aplicativos troquem dados e funcionalidades com facilidade e segurança. Uma API nada mais é que um conjunto de regras definidas que explicam como os computadores ou aplicativos se comunicam entre si. As APIs ficam entre um aplicativo e o servidor da web agindo como uma camada intermediária que processa a transferência de dados entre sistemas (IBM CLOUD EDUCATION, 2020).

Veja como funciona uma API:

1. Um aplicativo cliente inicia uma chamada de API, também conhecida como solicitação, para recuperar informações.
2. Depois de receber uma solicitação válida, a API faz uma chamada para o programa externo ou servidor web.
3. O servidor envia uma resposta à API com as informações solicitadas.
4. A API transfere os dados para o aplicativo solicitante inicial.

APIs oferecem segurança por design, porque sua posição como intermediário facilita a abstração de funcionalidades entre dois sistemas – o terminal da API separa o aplicativo de consumo da infraestrutura que fornece o serviço (IBM CLOUD EDUCATION, 2020).

Conceitos de componentes

Segundo Eler (2006), o desenvolvimento de software baseado em componentes concentra-se nas técnicas e práticas usadas para construir sistemas de software a partir de componentes preexistentes, adquiridos tanto por terceiros quanto pelo próprio desenvolvedor do sistema. Essa abordagem libera os programadores de pensarem sobre os detalhes de implementação para que possam se preocupar mais com o programa usado para compor os sistemas de software. O objetivo desse modelo de desenvolvimento é definir, qualificar, adaptar, integrar e atualizar componentes de software reutilizáveis, os quais vão desde controles de interface de usuário até componentes de entrega e domínio específico.

Diferenças entre frameworks, APIs e componentes

As diferenças existentes entre essas ferramentas são o ambiente de programação, o tipo de sistema a ser desenvolvido e quais as funcionalidades do software a ser criado. Com base nessas questões, podemos definir com sucesso o que podemos utilizar para melhorar o desempenho do desenvolvimento, podendo, assim, reutilizar determinados códigos ou estruturas. Nesse sentido, uma boa escolha pode impactar o desenvolvimento do projeto inteiro, fazendo com que o tempo para entrega ao cliente final seja bem menor.

VIDEOAULA: CONCEITOS DE FRAMEWORKS E A DIFERENÇA ENTRE FRAMEWORK, API E COMPONENTES

O vídeo mostrará mais detalhadamente os conceitos de cada um dos tópicos (framework, API e componente). Além disso, evidenciará quais vantagens e desvantagens há em cada modelo e o que implica a utilização de cada um no processo de construção de um software.

Videoaula: Conceitos de frameworks e a diferença entre framework, API e componentes

Para visualizar o objeto, acesse seu material digital.

APLICAÇÕES DE FRAMEWORKS E SEUS BENEFÍCIOS

Benefícios da utilização de frameworks

Devido à enorme variedade de linguagens de programação, há um grande número de frameworks disponíveis. Cada uma dessas estruturas é usada para resolver diferentes problemas que surgem no processo de desenvolvimento de software e são usadas em muitos setores, incluindo desenvolvimento da web, ciência de dados e muito mais.

O uso de frameworks economiza tempo e reduz o risco de erros, pois não é preciso escrever tudo do zero. Além disso, as estruturas já foram testadas, então há menos ainda com o que se preocupar. Outras vantagens incluem:

- Código mais seguro.
- Teste e depuração mais simples.
- Prevenção quanto a códigos duplicados.
- Código limpo e facilmente adaptável.
- Capacidade de se concentrar na escrita de código específico para o projeto.
- Possibilidade de extensão.

Para Ghimire (2020), a criação de aplicativos da web é um processo de desenvolvimento de aplicativos de software que pode ser executado em sites. Porém, embora siga o mesmo processo de desenvolvimento de software, é preciso pontuar que apresenta uma tecnologia e arquitetura bem diferentes. O aplicativo de software executado em um computador pessoal (PC) pode não depender da internet, ao contrário do aplicativo da web, que depende de servidores remotos. Os aplicativos da web e a tecnologia cliente-servidor avançaram bastante em comparação com aplicativos simples. Em consequência disso, hoje em dia, os aplicativos da web vêm em diferentes formas e tamanhos, como estático, dinâmico, sistema de

gerenciamento de conteúdo, *e-commerce* e jogos para portais de compartilhamento de conteúdo ao vivo. A tecnologia comumente compartilhada dos tipos de aplicativos anteriormente mencionados é a tecnologia de *back-end* e *front-end*.

Back-end web frameworks

Segundo Kaluža, Kalanj e Vukelić (2019), um aplicativo da web consiste em uma parte, a do cliente, que é executada no navegador da web do usuário, e outra, do servidor, que é executada em um servidor da web, o qual executa vários processos que criam um documento em formato HTML e o entregam ao cliente, que então interpreta e exibe o conteúdo HTML.

Assim, ao projetar um aplicativo da web, surge a questão sobre qual tecnologia escolher. O uso de framework no processo de desenvolvimento pode aumentar o nível de qualidade e padronização das aplicações web. Hoje, há uma variedade de frameworks no mercado que ajudam a criar aplicativos da web de várias maneiras.

O desenvolvimento de *back-end* lida com o lado lógico do aplicativo da web e diz respeito, principalmente, às linguagens de programação, à arquitetura central e às lógicas. Essas lógicas são escritas principalmente em linguagens de programação que podem ser executadas em servidores de computador e também influenciam em como os dados são armazenados, acessados e servidos a partir dos servidores.

Para cada framework existe uma linguagem de programação relacionada, de forma que são inúmeros os frameworks que podem ser utilizados para cada aplicação de determinada linguagem.

Front-end web frameworks

De acordo com Ghimire (2020), o desenvolvimento do *front-end* está mais voltado à estética e à parte de exibição do conteúdo, o que também é conhecido como desenvolvimento do lado do cliente. Um de seus maiores desafios é poder mostrar o material nos diferentes tipos de dispositivos e navegadores. Muitos dispositivos têm seu próprio *Software Development Kit* (SDK), que deve ser seguido para servir o mesmo conteúdo do navegador. O design do site, a usabilidade e a facilidade de uso são fatores essenciais abordados durante o desenvolvimento.

VIDEOAULA: APlicações DE FRAMEWORKS E SEUS BENEFÍCIOS

No vídeo serão abordados os conceitos de *back-end* e *front-end*, bem como suas tecnologias e as linguagens associadas à sua utilização. Serão apresentadas também algumas vantagens e desvantagens da utilização de determinadas ferramentas que estão no mercado, como os frameworks **Spring**, **Angular**, **JS** e outros.

Videoaula: Aplicações de frameworks e seus benefícios

Para visualizar o objeto, acesse seu material digital.

ESTUDO DE CASO

Imagine que você é um freelancer e que está negociando com uma empresa o desenvolvimento de um sistema, o qual será destinado para uso interno dos funcionários. Nesse sentido, foram repassados alguns pontos a serem analisados por você para que o sistema esteja de acordo com a necessidade da contratante.

Esse sistema deve conter uma página inicial com as seguintes informações: Sobre, Ajuda e Serviços. Todas elas serão repassadas pelo cliente, você apenas se preocupará com o layout e as rotinas.

Ainda, o sistema deve contar com o login, cadastro e visualização de relatórios gerenciais. Assim sendo, você precisa entender o que é *back-end*, o que é *front-end* e quais as respectivas linguagens e frameworks que você poderá utilizar. Apresente, neste caso de estudo, as diferenças entre as tecnologias de *back-end* e *front-end*, a linguagem a ser utilizada e pelo menos um framework que possa ajudar na construção desse sistema.

RESOLUÇÃO DO ESTUDO DE CASO

Como já apresentado durante nossa aula, a diferença entre *back-end* e *front-end* são bem claras quando se trata de aplicativos para web. O *back-end* diz respeito à parte da codificação do servidor, o que o sistema fará quando requisitado, uma rotina ou uma ação. Já o *front-end* é a parte visual do sistema e onde há interação com o usuário.

Por se tratar de um sistema simples, podemos pensar em uma linguagem como PHP para o desenvolvimento do *back-end*, ou, se preferir, a linguagem de programação Java também é uma ótima alternativa para codificação, pois ambas as linguagens são robustas e com boa aceitabilidade de mercado.

Quando nos referimos à utilização de frameworks, o ideal é analisar a sua necessidade em relação ao sistema. Se a linguagem escolhida for **PHP**, frameworks que podem ser utilizados são o **Angular**, o **Jquery**, entre inúmeros outros. Agora, se for a linguagem Java, pode-se contar com o **Spring**, o **JSF** e outros que são bem vistos no mercado.

Resolução do Estudo de Caso

Para visualizar o objeto, acesse seu material digital.

6º Saiba mais

A utilização de *frameworks* assume um papel muito importante no desenvolvimento de sistemas de um modo geral. Eles estão presentes em muitos sistemas conhecidos, como **Netflix**, **Amazon** e **Google**. Para conhecer um pouco mais sobre os frameworks utilizados por esses serviços, acesse os links indicados.

- https://docs.aws.amazon.com/index.html?nc2=h_qj_doc_do. Acesso em: 3 jan. 2022.
- <https://www.tecmundo.com.br/software/204629-desenvolvimento-app-escolhendo-linguagem-programacao.htm>. Acesso em: 3 jan. 2022.

Aula 2

CATEGORIAS DE FRAMEWORKS

Considerando que a utilização de frameworks é imprescindível em projetos de software, é necessário aprender corretamente seu uso para que a construção do programa, de fato, aconteça da melhor forma possível e causando o mínimo de erros durante a utilização pelo cliente.

40 minutos

INTRODUÇÃO

Dois dos objetivos dos frameworks são a organização e o gerenciamento bem-sucedidos para a construção de softwares diversificados, procedimentos que podem ser realizados de variadas formas e com inúmeras linguagens de programação.

Considerando que a utilização de frameworks é imprescindível em projetos de software, é necessário aprender corretamente seu uso para que a construção do programa, de fato, aconteça da melhor forma possível e causando o mínimo de erros durante a utilização pelo cliente.

Dessa forma, é importante que o estudo seja realizado com base nos materiais propostos e também em pesquisas que você, aluno, pode realizar. Foque nos exercícios e realize o máximo de leituras que puder sobre o assunto, sendo assim os resultados logo aparecerão. Bons estudos!

INTRODUÇÃO AS DIFERENTES CATEGORIAS DE FRAMEWORKS

Utilização de frameworks

Quanto ao processo de criação de software, pode-se dizer que ele é complexo e que requer muito trabalho, codificação, projeto e teste. Frameworks tornam a programação da web mais fácil e mais bem organizada de várias maneiras, além de aumentarem a produtividade da programação, uma vez que, com ele, centenas de linhas de código podem ser feitas em minutos, por meio de funções integradas do framework, fato que evita sua criação do zero, algo extremamente positivo, pois escrever uma única parte do código geralmente leva horas (PROKOFYEVA; BOLTUNOVA, 2017).

Para sistemas de computador, um framework é considerado uma estrutura que possui camadas e que demonstra como as aplicações devem ser construídas e como devem se relacionar. Geralmente esses frameworks oferecem interfaces prontas, ou seja, são projetos reutilizáveis para determinados tipos de softwares específicos.

Nesse sentido, existem diversas categorias de frameworks, as quais abrangem diversas áreas da computação para cada tipo de linguagem ou área relacionada para construção de vários softwares.

Categorias de frameworks

Para desenvolver um software, é preciso estar atento às estruturas que vão ao encontro das necessidades do programador. Para isso, é necessário entender o contexto do software e o objetivo final dele. Tendo isso em vista, pode-se encontrar os melhores frameworks do mercado, o que facilita a construção e a manutenção desses softwares no início e durante a construção e na manutenção após a conclusão (ZUSE, 2013).

Existem várias categorias em que podemos enquadrar os frameworks, e a escolha delas dependerá da finalidade para a qual o software será construído. Dentre os tipos existentes, destacam-se os que serão confeccionados para web e para utilização em *mobile* e *Big Data*, por exemplo.

Frameworks para aplicativos web

Uma estrutura da web é uma coleção de pacotes ou módulos que permitem aos desenvolvedores escreverem aplicativos da web ou serviços sem terem que lidar com detalhes de baixo nível como protocolos, *sockets* ou gerenciamento de processos/*threads* (WEB FRAMEWORKS, 2021).

A seguir são citados alguns exemplos dos frameworks mais conhecidos desenvolvidos para web:

- **Angular e Laravel**: para linguagem de programação **PHP**.
- **Spring e JSF**: para linguagem de programação **Java**.

Para cada linguagem de programação existem vários frameworks e para cada tipo de utilização há um específico.

Frameworks para aplicativos *mobile*

Um framework de desenvolvimento móvel é uma estrutura de software projetada para suportar o desenvolvimento de aplicativos móveis. É também uma biblioteca de software que fornece uma estrutura fundamental para apoiar o desenvolvimento de aplicações para um ambiente específico (FLUTTER, [s. d.]).

Existem diversos frameworks destinados para o desenvolvimento de aplicativos móveis, dentre eles destacam-se os seguintes:

- **Ionic**.
- **Flutter**.
- **Xamarin**.

Os programadores podem facilmente trabalhar com qualquer um deles e obter ótimos resultados em seus aplicativos, uma vez que todos têm funcionalidades similares e linguagens de programação diferenciadas.

Frameworks para aplicativos de *Data Science*

A ciência de dados é uma abordagem multidisciplinar para extrair percepções açãoáveis dos grandes e crescentes volumes de dados coletados e criados pelas organizações de hoje. Ela abrange a preparação de dados para análise e processamento, realizando análises avançadas de dados e apresentando os resultados para revelar padrões e permitir que as partes interessadas tirem conclusões informadas (IBM CLOUD EDUCATION, 2020).

Alguns frameworks se destacam nessa área:

- **Django**.
- **Pyramid**.
- **Scrapy**.

Essas estruturas são conhecidas e muito bem aceitas no mercado, tendo um resultado importante para *descoberta de padrões em Data Science*.

VIDEOAULA: INTRODUÇÃO AS DIFERENTES CATEGORIAS DE FRAMEWORKS

O vídeo a ser apresentado deve conter alguns dos frameworks que são utilizados no mercado, algumas de suas funcionalidades e quais empresas trabalham com eles, demonstrando exemplos importantes. Ainda, será mostrado à qual categoria cada um deles pertence.

Videoaula: Introdução as diferentes categorias de Frameworks

Para visualizar o objeto, acesse seu material digital.

FRAMEWORKS E AS LINGUAGENS DE PROGRAMAÇÃO

Etapas de desenvolvimento

Os frameworks são estruturas de código já escrito amplamente utilizados por milhares de programadores e que são bem aceitos em diversos ambientes de desenvolvimento. Para cada linguagem de programação, por exemplo, existem determinados *frameworks* para cada ação ou para cada módulo em relação ao projeto que precisa ser implementado.

Algumas linguagens de programação se destacam no mercado, como **Java**, **C#** e **Python**, as quais são usadas para criação de aplicativos de diferentes tipos e ramificações. Nesse sentido, existem as categorias de criação do software, ou de cada fase, por assim dizer.

Algumas das etapas de desenvolvimento mais importantes iniciam-se na fase de planejamento do software, passam pelo desenvolvimento propriamente dito e finalizam com os testes do programa. Algumas das seguintes etapas, ou as chamadas categorias, destacam-se:

- Containers.
- Persistência de dados.
- Gerenciadores de transações.
- Agendamento de tarefas.
- Validações.
- Chamadas assíncronas.
- Desenvolvimento web (*front-end*).
- Autenticação.
- Relatórios.
- Testes.

para cada uma dessas etapas de desenvolvimento existem diversos frameworks que foram escritos para auxiliar o programador durante a codificação. Geralmente os frameworks são criados para tarefas que precisam ser recriadas todas as vezes na construção de softwares. Por isso, cada linguagem de programação possui determinada quantidade de frameworks.

Frameworks para linguagem JAVA

Frameworks Java são grupos de códigos que já foram escritos por desenvolvedores para criar aplicações por meio da linguagem de programação. Eles são específicos para desenvolver aplicações de software e programas Java e incluem funções e classes predefinidas, que são utilizadas para processar, inserir e gerenciar hardware e interagir com o software do sistema (REDHAT, 2019).

Existem vários frameworks **Java** disponíveis, o que determina qual deles usar são as habilidades do desenvolvedor ou os requisitos da aplicação. Estes são alguns dos *frameworks open source* mais utilizados:

- **Quarkus:** framework Java de *stack* rico e compatível com infraestruturas nativas e de nuvem baseadas em microserviços.
- **Grails:** escrito na linguagem de programação **Groovy**, disponibiliza frameworks de aplicação web com recursos de exibição para plugins **CSS** e **HTML**.
- **Hibernate:** framework de mapeamento objeto-relacional (ORM) utilizado no gerenciamento de bancos de dados relacionais. Suporta execução de consultas SQL com auxílio das APIs da **Java Database Connectivity (JDBC)**.

- **Google Web Toolkit (GWT)**: conecta equipes de desenvolvimento de *front-end* às de *back-end*.

Frameworks para linguagem PHP

As estruturas de **PHP** fornecem aos desenvolvedores de software uma estrutura para a construção de aplicativos de software baseados em PHP. Como alguns frameworks são adequados para determinados projetos, isso significa que saber qual deles escolher pode envolver uma pesquisa cuidadosa.

Frameworks PHP são baseados na própria linguagem de programação PHP. Utilizar uma estrutura PHP significa menos código para escrever e menor diferença para iniciar um projeto. A linguagem PHP é uma tecnologia de *back-end* muito bem vista no mercado para criação de aplicativos da web. Podemos destacar alguns frameworks PHP:

- **Laravel**: possui diversos recursos e design elegante.
- **Laminas Project**: dispõe de uma arquitetura baseada em componentes.
- **CodeIgniter**: possui arquitetura MVC e flexibilidade.

Além disso, existem muitos outros frameworks, com inúmeras funcionalidades, da linguagem PHP.

VIDEOAULA: FRAMEWORKS E AS LINGUAGENS DE PROGRAMAÇÃO

O vídeo deverá mostrar algumas das principais etapas de desenvolvimento de um software bem como exemplos de aplicações que rodam os frameworks. Ainda, será mostrado um pouco sobre a linguagem **Java**, seus frameworks e os benefícios de sua utilização, bem como os frameworks de **PHP** e sua utilização.

Videoaula: Frameworks e as linguagens de programação

Para visualizar o objeto, acesse seu material digital.

PRINCIPAIS FERRAMENTAS DAS CATEGORIAS DE FRAMEWORKS

Ferramentas para desenvolvimento

Existem diversas ferramentas para o desenvolvimento que são utilizadas junto com os frameworks. Conheceremos, a partir de agora, quais são elas e como se dá sua instalação e sua configuração.

Ambiente de desenvolvimento Eclipse

O *Integrated Development Environment* (IDE), que significa, em português, Ambiente de Desenvolvimento Integrado, possui várias funcionalidades. A plataforma define o conjunto de estruturas e serviços comuns que formam coletivamente a infraestrutura necessária para suportar o uso do Eclipse como um modelo de componente, como uma *Rich Client Platform* (RCP) e como uma plataforma de integração de ferramentas abrangente. Esses serviços e estruturas incluem um modelo de interface de usuário de *workbench* padrão e um kit de ferramentas de *widget* nativo portátil; um modelo de projeto para gerenciamento de recursos; gerenciamento delta de recurso automático para compiladores e construtores incrementais; infraestrutura de depuração independente de linguagem; e infraestrutura para gerenciamento de recursos com versão multiusuário distribuído (BRESSON *et al.*, 2016).

O IDE **Eclipse** é famoso por ser um ambiente para desenvolvimento da linguagem de programação **Java**, porém há outros IDEs que suportam outras linguagens, como IDE **C/C++**, IDE **JavaScript**, IDE **PHP** e outros. Nesse cenário, algumas plataformas se destacam:

- **Eclipse**: plataforma de uso e ferramentas.
- **Orion**: plataforma de integração de ferramenta aberta, baseada em navegador e totalmente focada no desenvolvimento web.
- **EclipseChe**: para o desenvolvimento de *kubernetes* (containers) para desenvolvedores.
- **Theia**: plataforma de nuvem e desktop para desenvolvimento web.

Ambiente de desenvolvimento NetBeans

O **NetBeans** IDE é um ambiente de desenvolvimento integrado de código aberto e também gratuito. O IDE simplifica o desenvolvimento de aplicativos para web, corporações, desktop e aparelhos móveis que utilizam as plataformas **Java** e **HTML5**, além de oferecer suporte para o desenvolvimento de aplicativos **PHP** e **C/C++**.

(NETBEANS IDE, c2022).

Ambiente de desenvolvimento Anaconda

Anaconda é uma distribuição das linguagens de programação **Python** e **R** para computação científica (ciência de dados, aplicativos de aprendizado de máquina, processamento de dados em larga escala, análise preditiva, etc.), que visa simplificar o gerenciamento e a implantação de pacotes. A distribuição inclui pacotes de ciência de dados adequados para **Windows**, **Linux** e **macOS** (ANACONDA, c2022).

A distribuição do **Anaconda** conta com mais de 250 pacotes instalados automaticamente, e mais de 7.500 pacotes adicionais de código aberto podem ser instalados a partir do gerenciador. Além disso, há ainda o pacote conda e o gerenciador de ambiente virtual. Ele também inclui uma interface, **Anaconda Navigator**, como uma alternativa gráfica para a interface de linha de comando (CLI) (ANACONDA, c2022).

Ambiente de desenvolvimento Visual Studio Code

O **Visual Studio Code** é um software editor de código-fonte que foi desenvolvido pela **Microsoft**. Ele inclui versionamento *Git* integrado, suporte à depuração, verificação de sintaxe, conclusão de código inteligente, refatoração de código, entre inúmeras outras ferramentas e funcionalidades. Por ser personalizável, permite que os usuários alterem temas, atalhos e opções dentro do ambiente. É um software gratuito e de código aberto (CODE, c2022).

O **Visual Studio Code** suporta um grande número de linguagens de programação e um conjunto de funcionalidades que estão disponíveis dependendo da linguagem. Dentre elas, podemos citar algumas como: **JavaScript, Python, C e C++, TypeScript, C#, HTML, CSS e Json** (CODE, c2022).

VIDEOAULA: PRINCIPAIS FERRAMENTAS DAS CATEGORIAS DE FRAMEWORKS

O vídeo a ser apresentado deve destacar as ferramentas apresentadas ao longo da aula. Nele serão apresentadas a interface de cada uma delas e a utilização da ferramenta relacionada com a linguagem de programação que ela suporta. Dentre as ferramentas apresentadas, estarão a IDE **Netbeans** e **Visual Studio Code**.

Videoaula: Principais ferramentas das categorias de frameworks

Para visualizar o objeto, acesse seu material digital.

ESTUDO DE CASO

Imagine que você trabalha em uma empresa de tecnologia e que lhe solicitaram a construção uma aplicação web. Você terá que apresentar as tecnologias a serem utilizadas e como vão auxiliar na construção dessa aplicação.

Nesse sentido, você deve mostrar quais tecnologias *back-end* deverão ser selecionadas e quais as tecnologias *front-end* serão aplicadas. É necessário descrever o porquê da utilização das tecnologias em questão, tendo em vista que o seu líder imediato avaliará os recursos e se você está atualizado com o mercado para que a construção da aplicação seja feita com sucesso.

RESOLUÇÃO DO ESTUDO DE CASO

Para a resolução desse estudo de caso, é necessário que você, aluno, pense no passo a passo para a construção do sistema em questão. Por se tratar de uma aplicação web, é necessário buscar e pesquisar as ferramentas disponíveis e as mais usuais de mercado. A partir disso, vamos tratar cada uma delas separadamente, dividindo-as em dois momentos.

No primeiro momento, é necessário decidir que linguagem será utilizada, podendo ela ser **Java, Python, PHP** ou qualquer outra cujo desenvolvimento seja voltado para web. Nesse exemplo, utilizaremos a linguagem de programação **PHP**, que é de alto nível, código aberto, com compatibilidade com inúmeros bancos de dados e que também pode ser facilmente hospedada em vários servidores. Dessa forma, a aplicação poderá facilmente ser construída e hospedada em nuvem ou localmente.

No segundo momento, é importante utilizar frameworks compatíveis com a linguagem de programação escolhida, neste caso, o **PHP**. Vimos durante a aula que existem inúmeros frameworks que podem auxiliar, e muito, a construção de sistemas, sendo assim, neste exemplo, utilizaremos para *front-end* frameworks como

Bootstrap ou **Materialize**, que são ótimas alternativas; já para o *back-end* podemos pensar em frameworks como o **Laravel** ou o **CodeIgniter** que são bem específicos para criação MVC e de banco de dados.

Resolução do Estudo de Caso

Para visualizar o objeto, acesse seu material digital.

6º Saiba mais

Para conhecer um pouco mais sobre o que são e como funcionam os ambientes de desenvolvimento, visite o sítio a seguir:

<https://www.oracle.com/br/tools/technologies/netbeans-ide.html>. Acesso em: 12 jan. 2022.

Aula 3

CASES DE FRAMEWORKS EM JAVA, JAVASCRIPT E PHP

O objetivo desta aula é que você, aluno, entenda quais são as ferramentas mais empregadas no mercado de trabalho, que aprenda a utilizá-las e que esteja preparado para qualquer desafio que possa surgir ao longo de sua caminhada.

39 minutos

INTRODUÇÃO

O objetivo desta aula é que você, aluno, entenda quais são as ferramentas mais empregadas no mercado de trabalho, que aprenda a utilizá-las e que esteja preparado para qualquer desafio que possa surgir ao longo de sua caminhada.

Você deverá estudar sobre as linguagens de programação e as ferramentas básicas de cada uma delas e será convidado a acompanhar trechos de códigos para memorização e fixação daquilo que será passado.

Dedique um tempo para pesquisa e autoestudo, busque os termos que não estejam bem claros para você, com isso a memorização e o entendimento com certeza serão bem maiores. Bons estudos!

JAVA: CASES DE EXEMPLOS BÁSICOS, NETBEANS E ECLIPSE

Linguagem de programação Java

O **Java** é uma linguagem de programação e uma plataforma de computador lançada pela empresa **Sun Microsystems** no ano de 1995. Existem muitos aplicativos e sites que só funcionarão se o **Java** estiver instalado, e muitos outros estão sendo criados dentro dessas condições diariamente. **Java** é uma plataforma rápida, segura e confiável. De laptops a *data centers*, consoles de jogos a supercomputadores científicos, telefones celulares à internet, o **Java** está em toda parte (JAVA, [s. d.]).

Sobre a programação Java

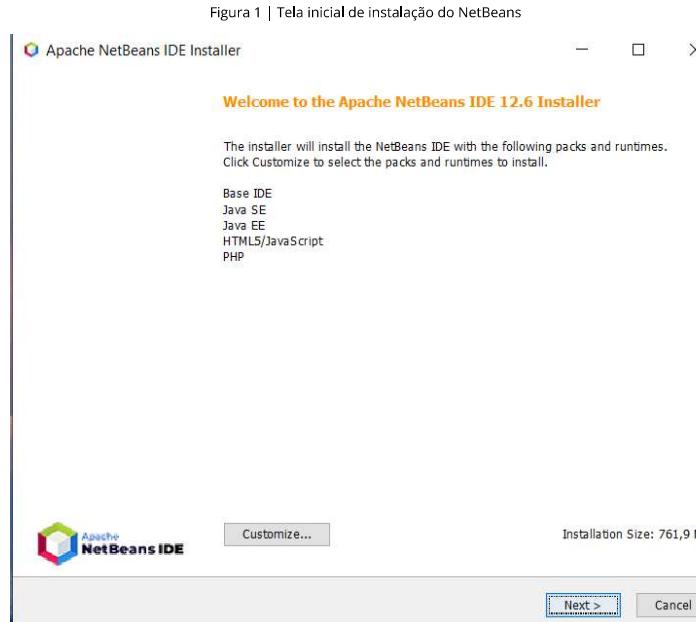
Tendo em vista sua importância, é relevante pontuar algumas características sobre a programação com a linguagem **Java**:

- **Independente de plataforma** – Podemos escrever código **Java** em uma plataforma (sistema operacional) e rodar em outra sem nenhuma modificação.
- **Orientada a objetos** – Java é uma linguagem orientada a objetos, o que ajuda a tornar nosso código Java mais flexível e reutilizável.
- **Veloz** – Um código Java bem otimizado é quase tão rápido quanto linguagens de nível inferior, como **C++**, e muito mais rápido que **Python**, **PHP**, etc.

Instalação do IDE NetBeans para programação em Java

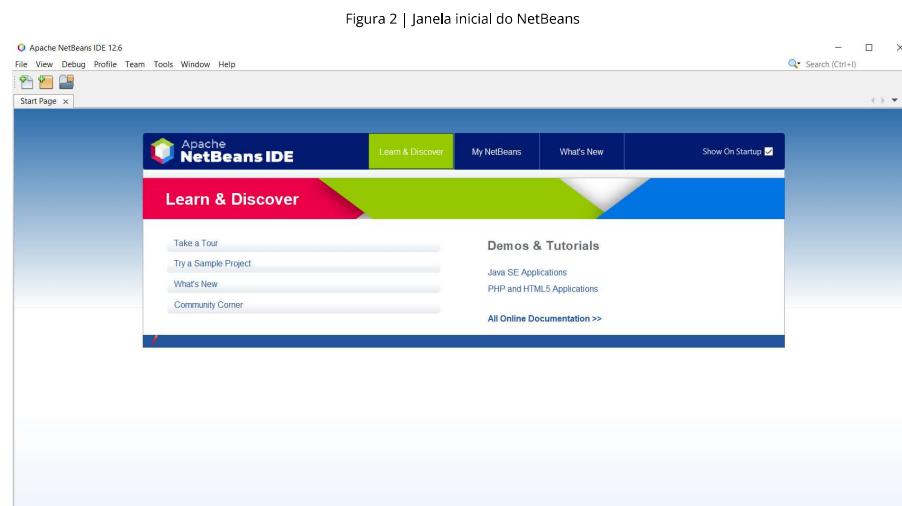
O **NetBeans IDE** (*Integrated Development Environment*) é um ambiente de desenvolvimento integrado de código aberto e gratuito. Para utilizá-lo basta acessar o site através do link a seguir e fazer o download da versão atualizada: <https://www.oracle.com/br/tools/technologies/netbeans-ide.html>.

Depois do download, basta executar o arquivo e seguir o passo a passo da instalação como mostra a Figura 1.



Fonte: captura de tela de NetBeans IDE.

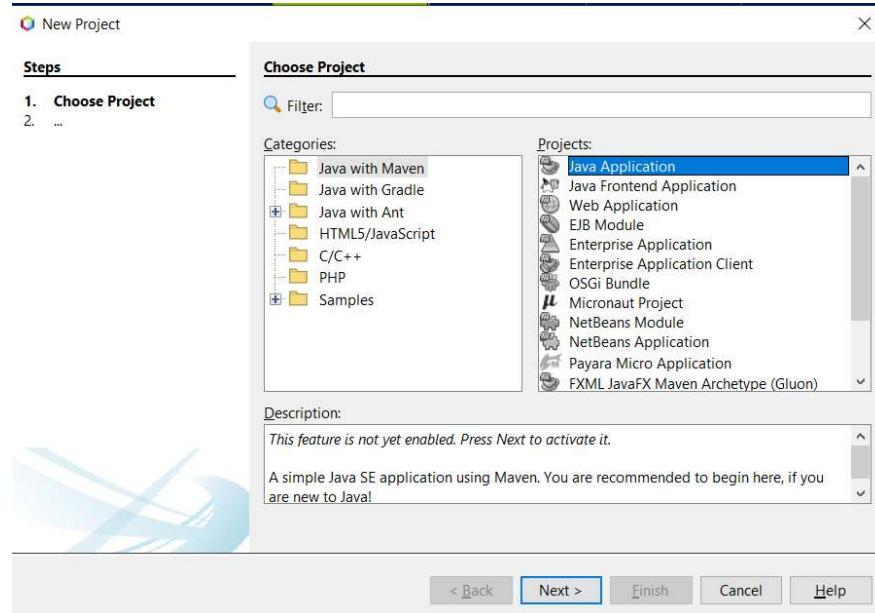
Após feita a instalação, podemos executar o programa. A Figura 2 mostra como deve aparecer a janela para execução.



Fonte: captura de tela de NetBeans IDE.

Nesse sentido, podemos criar um novo projeto Java, seguindo os passos: 'File >> New Project'. Feito isso, a seguinte tela surgirá:

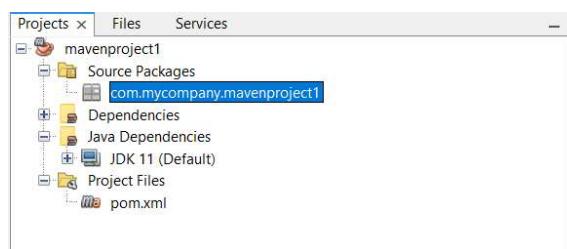
Figura 3 | Tela de escolha de projeto



Fonte: captura de tela de NetBeans IDE.

Para seguir criando nosso primeiro projeto, basta aplicar as configurações, e uma estrutura será criada automaticamente conforme a Figura 4.

Figura 4 | Estrutura gerada pelo NetBeans



Fonte: captura de tela de Painel de Controle Windows.

Após a criação da estrutura, é só criar uma classe e desenvolver um código básico de Java. O exemplo a seguir mostra o primeiro programa em **Java**.

Figura 5 | Exemplo do código Java

```
NewClass.java x
Source History
1 /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to edit this
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit thi
4  */
5 package com.mycompany.mavenproject1;
6
7 /**
8  * @author
9  */
10 public class NewClass {
11
12     public static void main(String[] args) {
13         System.out.println("Olá Mundo");
14     }
15
16 }
17 }
```

Fonte: captura de tela de NetBeans IDE.

Instalação do IDE Eclipse para programação em Java

O **IDE Eclipse** é um ambiente de programação similar ao **NetBeans**, porém traz recursos diferentes. O download dele pode ser realizado por meio do seguinte link: <https://www.eclipse.org/downloads/>.

Assim como o NetBeans, essa ferramenta de desenvolvimento é gratuita. Com isso, após o download, basta executá-la e instalá-la. Depois de aceitar os termos e realizar a instalação, uma janela de inicialização será aberta, tal como mostrado na Figura 6.

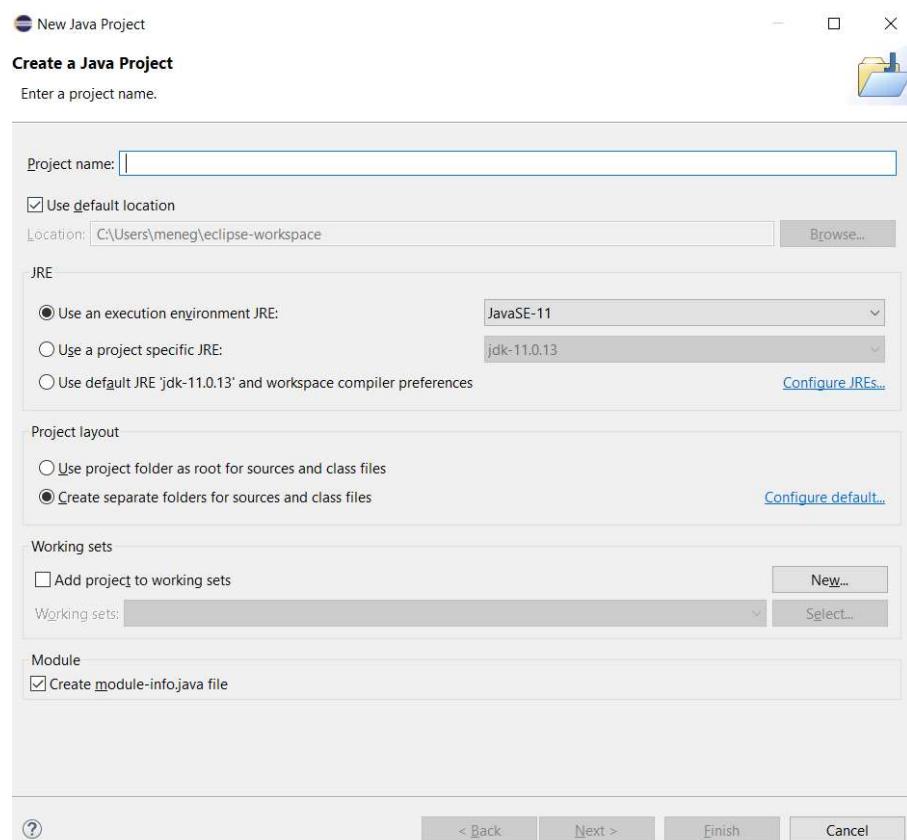
Figura 6 | Tela inicial do Eclipse



Fonte: captura de tela de IDE Eclipse.

Para criar um projeto Java na IDE, clique no menu '*File* >> *New Java Project*', desse modo, uma nova tela abrirá mostrando as opções para criação do projeto.

Figura 7 | Tela de criação do projeto Eclipse



Fonte: captura de tela de IDE Eclipse. Elaborada pelo autor.

Com isso, uma estrutura básica será criada. A partir desse ponto, crie um pacote e uma classe no menu '*File* >> *New Package*' e posteriormente '*File* >> *New Class*'. Para criar um código de teste, podemos utilizar o mesmo exemplo feito no **NetBeans**. Após esse passo a passo, é só executar, e, assim, teremos nosso primeiro programa em **Java** criado em ambas as plataformas.

VIDEOAULA: JAVA: CASES DE EXEMPLOS BÁSICOS, NETBEANS E ECLIPSE

No vídeo sobre o bloco, além de ser realizada a instalação de ambas as ferramentas (**NetBeans** e **Eclipse**), será mostrado um passo a passo da montagem de um projeto piloto nessas ferramentas, o que fará com que o aluno entenda um pouco mais a estrutura de cada uma das ferramentas.

Videoaula: Java: cases de exemplos básicos, NetBeans e Eclipse

Para visualizar o objeto, acesse seu material digital.

JAVASCRIPT: CASES UTILIZANDO O NETBEANS E O WEBSTORM

Visão geral de JavaScript

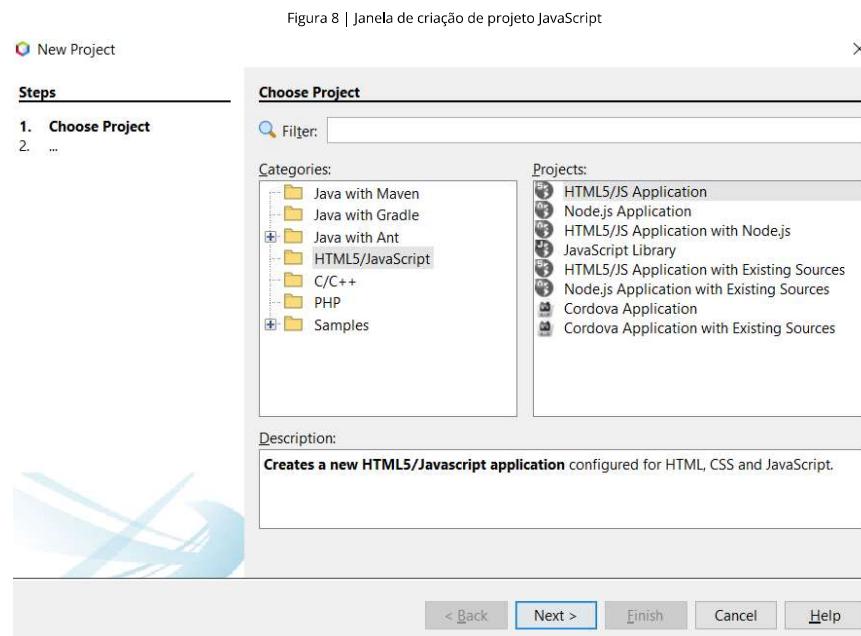
JavaScript é uma linguagem multiparadigma, dinâmica, com tipos e operadores, com objetos padrão embutidos e métodos. Sua sintaxe é baseada nas linguagens **Java** e **C** – muitas estruturas dessas linguagens também se aplicam ao **JavaScript**, que suporta programação orientada a objetos com protótipos de objetos em vez de classes. Além disso, o JavaScript também suporta programação funcional, uma vez que, por serem objetos, as funções podem ser armazenadas em variáveis e passadas como qualquer outro objeto. Os programas JavaScript manipulam valores, os quais pertencem todos a um só tipo. Os tipos de **JavaScript** são: *Number, BigInt, String, Boolean, Function, Object*, entre outros (MDN, 2022).

O JavaScript foi inicialmente criado para “tornar as páginas da web vivas”. Os programas nessa linguagem são chamados de scripts; eles podem ser escritos diretamente no **HTML** de uma página da web e executados automaticamente à medida que a página é carregada. Os scripts são fornecidos e executados como texto simples, pois não precisam de preparação ou compilação especial para serem executados. Nesse aspecto, o JavaScript é muito diferente de outra linguagem chamada Java (JAVASCRIPT, 2021).

Cases de JavaScript utilizando NetBeans

Como já vimos em outros blocos, o download e a instalação do NetBeans podem ser facilmente realizados por meio do acesso ao site: <https://netbeans.apache.org/download/index.html>.

Para executar um projeto com JavaScript é necessário abrir o NetBeans, clicar em ‘File >> New Project’ e selecionar o tipo ‘**HTML5/JavaScript**’.



Fonte: captura de tela de NetBeans IDE.

Feito isso, avance os próximos passos e será criada automaticamente a estrutura do projeto. Um arquivo chamado ‘index.html’ será criado e, dentro dele, poderemos inserir códigos **JavaScript** usando a tag ‘**<script>** **</script>**’. No meio dessa tag inserimos os comandos ou, então, podemos criar um arquivo do tipo **.js**, que é próprio de execução do **JavaScript**. Um exemplo pode ser criado e executado para testes conforme Figura 9.

Figura 9 | Exemplo HTML e JavaScript

```

<!DOCTYPE html>
<html>
<body>
<h2>What Can JavaScript Do?</h2>
<p id="demo">JavaScript can change HTML content.</p>
<button type="button" onclick="document.getElementById('demo').innerHTML = 'Hello JavaScript!'">Click Me!</button>
</body>
</html>

```

Fonte: captura de tela de JavaScript.

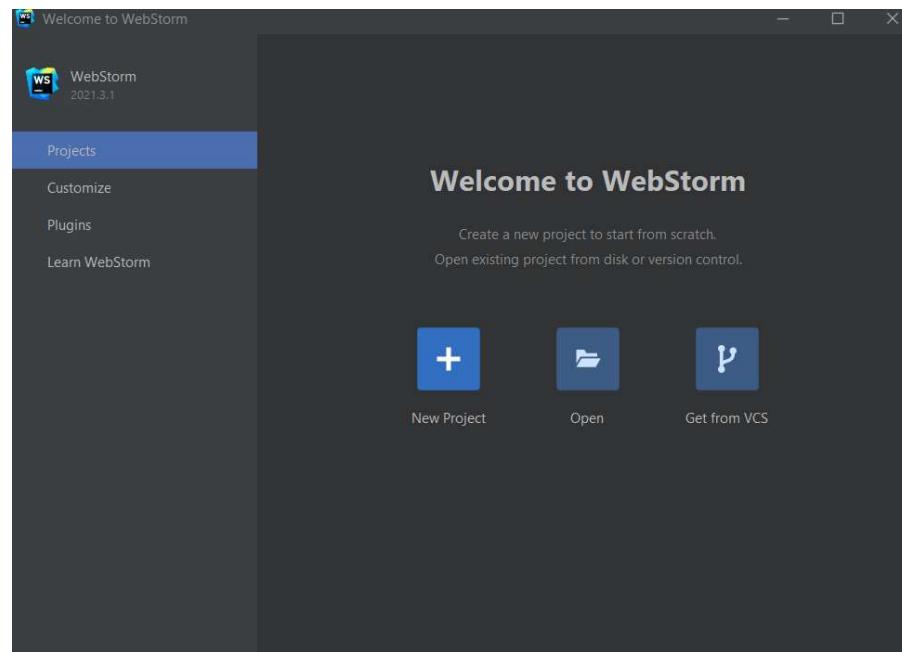
Cases de JavaScript utilizando WebStorm

O **IDE WebStorm** é um ambiente de desenvolvimento integrado para **JavaScript** e várias tecnologias integradas. A empresa que desenvolve essa ferramenta é a **JetBrains**, a qual torna a experiência de programação mais agradável e automatiza o trabalho de rotina transformando atividades complexas em atividades mais simples (JETBRAINS, c2022).

Para instalar essa **IDE**, basta realizar o download gratuito da ferramenta, acessando o site:
<https://www.jetbrains.com/pt-br/webstorm/>.

Após download, execute e instale o programa no ambiente em que desejar, seja **Windows**, **Linux** ou **Mac**. O ambiente a ser mostrado após instalação é o que aparece na Figura 10.

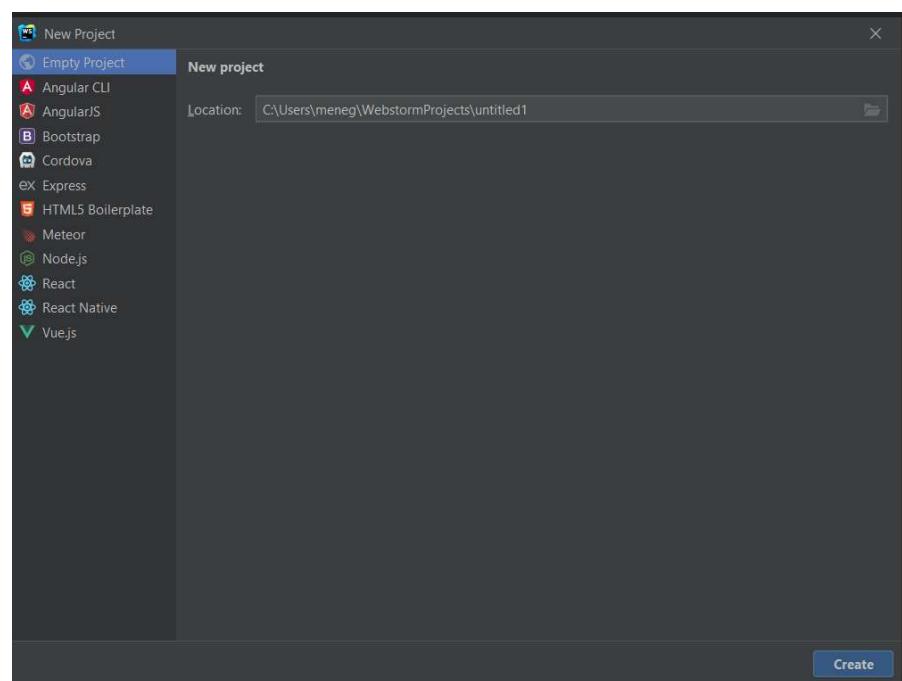
Figura 10 | Tela inicial do JetBrains



Fonte: captura de tela de JetBrains.

Pode-se criar um novo projeto em *New Project*. Após clicar no ícone, você será direcionado à tela principal do programa, onde poderá criar um projeto de diversos tipos, dentre eles, projetos embutidos com JavaScript.

Figura 11 | Tela de criação de Projeto JetBrains



Fonte: captura de tela de JetBrains.

Para executar um código JavaScript, basta criar um arquivo do mesmo tipo e inserir os códigos, com os quais poderá ser executado e os quais tornarão possível a verificação de seu funcionamento, como ocorre no exemplo adiante.

Figura 12 | Exemplo de código JavaScript

```
<script>
let x = 5;
let y = 6;
let z = x + y;
document.getElementById("demo").innerHTML =
"The value of z is: " + z;
</script>
```

Fonte: elaborada pelo autor.

VIDEOAULA: JAVASCRIPT: CASES UTILIZANDO O NETBEANS E O WEBSTORM

O vídeo deste bloco contempla as ferramentas para construção de aplicativos básicos e também de *cases* simples para utilização com a linguagem JavaScript. Ainda, serão mostrados os exemplos em funcionamento, evidenciando, assim, a importância de cada ambiente de desenvolvimento e de suas ferramentas.

Vídeoaula: JavaScript: cases utilizando o NetBeans e o WebStorm

Para visualizar o objeto, acesse seu material digital.

PHP: CASES UTILIZANDO O NETBEANS E O VISUAL STUDIO

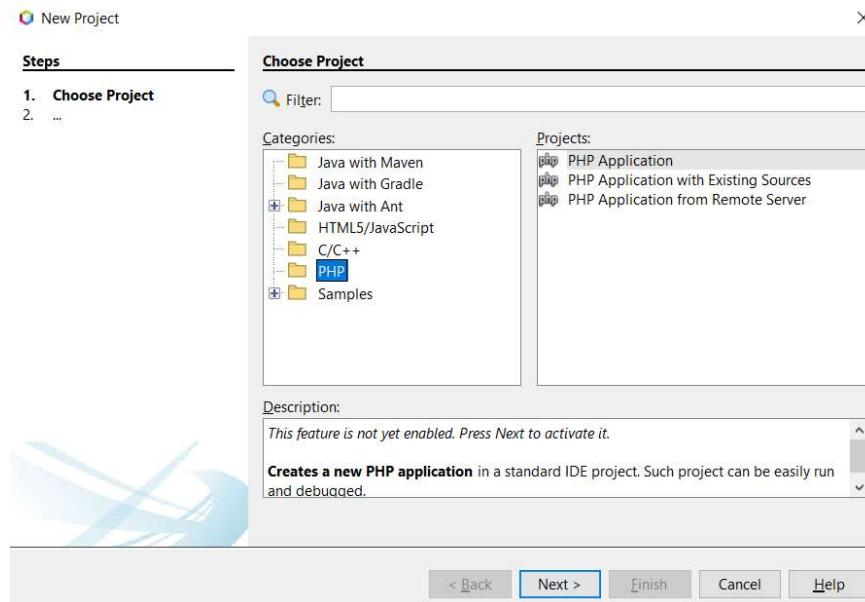
O que é o PHP?

Para Niederauer (2017), o *Hypertext Preprocessor* (PHP) é uma linguagem de script do lado do servidor gratuita e de código aberto amplamente utilizada no desenvolvimento web. Qualquer pessoa pode se beneficiar por aprender PHP, mas é ainda mais essencial para aqueles interessados em programação web. O PHP está disponível em todos os principais sistemas operacionais, como **Linux**, **Microsoft Windows** e **macOS**. A maioria dos servidores web, incluindo **Apache** e **IIS**, também suporta PHP.

Cases PHP utilizando NetBeans

Para utilizar o PHP com o NetBeans, basta instalar e executar este último, como mostrado nos blocos anteriores. Nesse sentido, para criar um projeto inicial, basta clicar no menu '*File >> New Project >> PHP*', como mostrado na Figura 13.

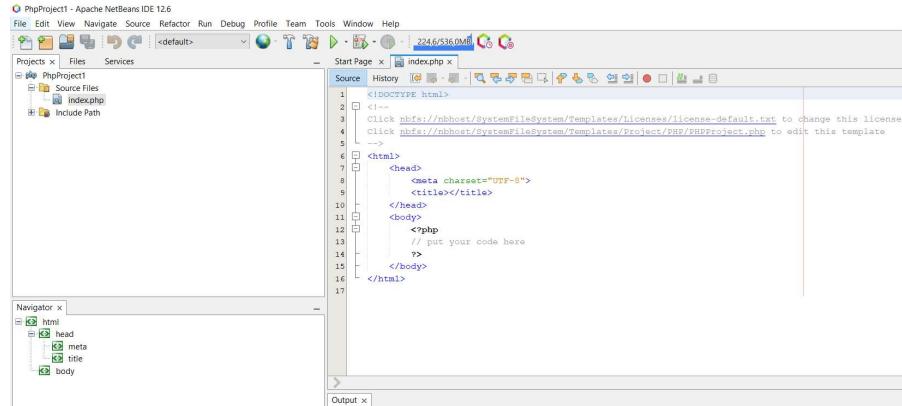
Figura 13 | Tela de Escolha de Projeto PHP no NetBeans



Fonte: captura de tela da NetBeans IDE.

Após avançar as etapas da janela, pode-se selecionar a configuração desejada de outros frameworks à sua escolha. O projeto seguirá com a seguinte estrutura:

Figura 14 | Estrutura do projeto no NetBeans



Fonte: captura de tela de NetBeans IDE.

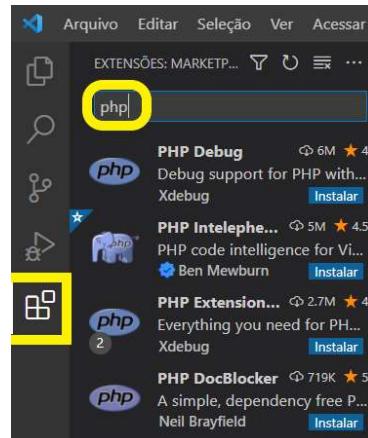
Cases PHP utilizando Visual Studio

O **Visual Studio Code** é um editor de código-fonte leve e poderoso, que é executado na área de trabalho e está disponível para **Windows, macOS e Linux**. Ele vem com suporte embutido para **JavaScript, TypeScript e Node.js** e possui um rico ecossistema de extensibilidade para outras linguagens (como **C, C#, Java, Python, PHP, Go**) e *runtimes* (como **.NET e Unity**) (a.CODE, c2022).

Para a instalação, basta acessar o site indicado a seguir e fazer o download: <https://code.visualstudio.com/>.

Feitos o download e a instalação, a janela principal deverá ser aberta. Nela, você terá opções de escolha de tema e *plugins* de linguagens que poderá utilizar. No caso, selecionaremos o PHP. No menu extensões, procuraremos por "PHP" e logo aparecerão várias extensões relacionadas. Também existem diversas outras que podem ser adicionadas conforme a necessidade do programador.

Figura 15 | Extensões PHP



Fonte: captura de tela adaptada de Visual Studio Code.

Após a instalação do **Visual Studio Code**, é necessário fazer também o download do PHP executável para que os projetos sejam compilados, o que é bem simples. A página para download está disponível em: <https://windows.php.net/download#php-8.1>.

Depois de fazer o download para a plataforma escolhida, basta descompactar o arquivo e configurar o Visual Studio Code. Para seguir um passo a passo detalhado, utilize o link disponível em: <https://code.visualstudio.com/docs/languages/php>.

Para executar um arquivo nos ambientes, podemos inserir um trecho de código PHP para testar o funcionamento dele, assim como mostrado na Figura 16.

Figura 16 | Trecho de código teste em PHP

```
<?php
echo "Hello world";
// ... mais código
echo "última instrução";
// o script termina aqui, sem tag de fechamento PHP
```

Fonte: elaborada pelo autor.

VIDEOAULA: PHP: CASES UTILIZANDO O NETBEANS E O VISUAL STUDIO

O vídeo deverá mostrar um pouco mais sobre as ferramentas **Visual Studio Code** e sobre o **NetBeans**. Serão criados alguns trechos de códigos de **JavaScript** e de **PHP** e, por fim, será explanado como funcionam algumas das ferramentas que são nativas dos **IDEs**.

Videoaula: PHP: Cases utilizando o NetBeans e o Visual Studio

Para visualizar o objeto, acesse seu material digital.

ESTUDO DE CASO

Caro aluno, imagine-se trabalhando em um ambiente corporativo onde você foi escolhido para apresentar um relatório com algumas ferramentas aos gerentes de tecnologia, que precisam conhecer melhor as características de determinadas linguagens de programação e de ferramentas para programação em geral.

Tendo em vista esse cenário, você precisa criar um documento que descreva as linguagens em alta no mercado, que estão sendo bem aceitas e que tenham perspectiva de crescimento. O seu objetivo é mostrar quais as linguagens mais seguras e mais prósperas. Você também deve demonstrar quais ferramentas pode utilizar, por isso dê prioridade para as gratuitas, pois a economia também é uma característica importante para a empresa.

RESOLUÇÃO DO ESTUDO DE CASO

Caro aluno, o seu relatório deve contar com algumas linguagens já conhecidas, mas também pode lançar mão de pesquisas que introduzam novas tecnologias presentes no mercado, a fim de não se prender apenas a linguagens já conhecidas, como **Java**, **C++** ou **.Net**, pois existem outras como **Python**, **Kotlin**, **Go** e outras mais.

Nesse sentido, em relação às ferramentas, podemos apresentar as diversos trabalhadas em aula, como **Visual Studio Code**, **NetBeans** e, claro, **Eclipse**. Existem outros ambientes que também se destacam e que devem ser lembrados, como os de programação colaborativos na web e utilização de ferramentas, como o **GIT**. Fique atento sempre às novidades, pois assim será fácil construir qualquer aplicativo utilizando o mínimo de recursos possível.

Resolução do Estudo de Caso

Para visualizar o objeto, acesse seu material digital.

6º Saiba mais

Você poderá se aprofundar por meio da leitura dos manuais completos das linguagens de programação PHP e Java, por exemplo, que estão disponíveis nos próprios sítios dessas linguagens. Para auxiliá-lo os links seguintes levam para ambos os sites dos manuais. Boa sorte e boa leitura dos conteúdos!

Java

Disponível em: <https://docs.oracle.com/en/java/>. Acesso em 21 jan. 2022.

PHP

Disponível em: <https://www.php.net/>. Acesso em: 21 jan. 2022.

Aula 4**CASES DE FRAMEWORK EM PYTHON, .NET E MOBILE**

Nesta aula abordaremos tanto linguagens de programação quanto ferramentas muito usuais para o mercado de trabalho e para todo o ambiente de desenvolvimento.

43 minutos

INTRODUÇÃO

Caro aluno, nesta aula abordaremos tanto linguagens de programação quanto ferramentas muito usuais para o mercado de trabalho e para todo o ambiente de desenvolvimento. Com esse estudo você poderá se aprofundar e poderá construir um conhecimento sólido. As linguagens de programação como **Python**, **.NET** e **Java** devem nortear seus estudos junto com ferramentas de desenvolvimento como **PyCharm**, **Visual Studio** e **Android Studio**.

Nesta aula, portanto, você deverá aprender sobre frameworks e tecnologias que se aplicam a eles e a suas linguagens. Criando exemplos práticos e reais, você deverá aprender com maior facilidade. Além disso, você também deve se dedicar à pesquisa para que todo conhecimento adquirido seja acrescentado aos apresentados em aula. Bons estudos!

PYTHON: CASES UTILIZANDO PYCHARM E ANACONDA**Programação em Python**

As pessoas usam **Python** em muitos lugares: sua rica biblioteca de base o torna excelente para todos os tipos de pequenos scripts auxiliares, além de ser dimensionado também para grandes sistemas. Para ilustrar isso, veja o exemplo dos criadores originais do **YouTube**, que usavam Python na maior parte do tempo. Você pode usar o Python, por exemplo, para automatizar tarefas, realizar cálculos, criar interfaces de usuário, criar *back-ends* de sites, acessar bancos de dados, baixar informações da internet, etc. É uma linguagem versátil, fácil de aprender e escrever. É perfeito para programadores iniciantes e é igualmente útil e poderoso para profissionais experientes.

Instalando Python

Pode-se instalar o Python de diversas formas, já que ele é multiplataforma. Sendo assim, iniciaremos com a instalação para o ambiente **Windows**. A **Microsoft** hospeda uma versão da comunidade do Python na **Microsoft Store**. Essa é a maneira recomendada de instalar o Python no Windows porque ele lida com atualizações automaticamente e também pode ser desinstalado facilmente.

Você também pode baixar um instalador do Python em seu site oficial para download. Esse método não fornece atualizações automáticas e é recomendado apenas se você não tiver acesso à loja da Microsoft. Ao usar esse instalador, certifique-se de marcar a caixa de seleção que diz "Add Python to PATH".

Figura 1 | Instalador Python



Fonte: Baaren (2022, [s. p.]).

Para instalação em outros sistemas operacionais acesse: <https://python.land/installing-python>.

Instalando PyCharm

PyCharm é um ambiente de desenvolvimento integrado (**IDE**) utilizado para programação na linguagem Python. Desenvolvido pela **JetBrains**, possui várias ferramentas de depuração, testes integrados e sistemas de controle de versão. O **PyCharm** possui versão proprietária e outra gratuita chamada de **PyCharm Community**.

Community. Para fazer o download da ferramenta basta acessar o sitio seguinte:

<https://www.jetbrains.com/pt-br/pycharm/download/#section=windows>.

Após download e execução do arquivo, basta aceitar os termos e executar o programa. Caso não tenha instalado o Python, o próprio software faz o download e configura automaticamente o ambiente para você. Para executar, basta pressionar as teclas “*shift + F10*”. Com isso, a execução do primeiro código será mostrada na tela após execução.

Figura 2 | Execução do primeiro programa em Python

The screenshot shows the PyCharm IDE interface. On the left, there's a project tree with a file named 'main.py'. The main editor window contains the following Python code:

```

1 # This is a sample Python script.
2
3 # Press Shift+F10 to execute it or replace it with your code.
4 # Press Double Shift to search everywhere for classes, files, tool windows, actions, and settings.
5
6
7 def print_hi(name):
8     # Use a breakpoint in the code line below to debug your script.
9     print(f'Hi, {name}')  # Press Ctrl+F8 to toggle the breakpoint.
10
11
12 # Press the green button in the gutter to run the script.
13 if __name__ == '__main__':
14     print_hi('PyCharm')
15
16 # See PyCharm help at https://www.jetbrains.com/help/pycharm/
17
18 print_hi()

```

At the bottom, the 'Run' tab shows the command: 'C:\Users\meneg\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/meneg/PycharmProjects/pythonProject/main.py' and the output: 'H1_PyCharm!'. A yellow box highlights the 'H1_PyCharm!' entry in the dropdown menu. The status bar at the bottom says 'Process finished with exit code 0'.

Fonte: captura de tela adaptada de Pycharm.

Instalação do *toolkit Anaconda*

Anaconda é um kit de ferramentas para ciência de dados. É muito conhecido na comunidade, tem código aberto e possui bibliotecas e pacotes incríveis de código aberto para realizar ciência de dados utilizando **Python/R** e aprendizado de máquina (ANACONDA, c2022).

Anaconda é uma plataforma muito popular, possui repositórios baseados em nuvem e nela existem mais de 7500 pacotes de ciência de dados e aprendizado de máquina. Possui também uma interface de navegação e gerenciamento dos pacotes chamada **Anaconda Navigator**. Pode ser utilizada também através de linha de comando (ANACONDA, c2022).

VIDEOAULA: PYTHON: CASES UTILIZANDO PYCHARM E ANACONDA

O vídeo deverá mostrar um pouco da instalação do **Python** e de como fazer o download da ferramenta. Além disso, serão mostrados o download e a instalação do **PyCharm** e suas principais características e ferramentas. Após isso, o *toolkit* Anaconda será executado e deverá ser mostrada a instalação de pacotes dentro do ambiente.

Videoaula: Python: cases utilizando PyCharm e Anaconda

Para visualizar o objeto, acesse seu material digital.

.NET: CASES UTILIZANDO O VISUAL STUDIO

O que é .NET?

.NET é uma plataforma de desenvolvimento de código aberto, criada pela Microsoft, que pode construir muitos tipos e diferentes aplicativos. Com o .NET, podemos usar várias linguagens com diversos editores e muitas bibliotecas para criar aplicativos para web, dispositivos móveis, desktops, jogos e **IoT (Internet of Thing/Internet das Coisas)**. Com isso, há possibilidade de escrever aplicativos .NET em **F#, C# ou Visual Basic** (MICROSOFT, c2022).

Sobre as linguagens aceitas pela plataforma, podemos especificá-las um pouco mais a seguir:

- C# é uma linguagem de programação moderna, simples, orientada a objetos e segura.
- F# é uma linguagem de programação de fácil escrita, robusta e com alto desempenho.
- O Visual Basic é uma linguagem fácil e com uma sintaxe simples para criar aplicativos orientados a objetos.

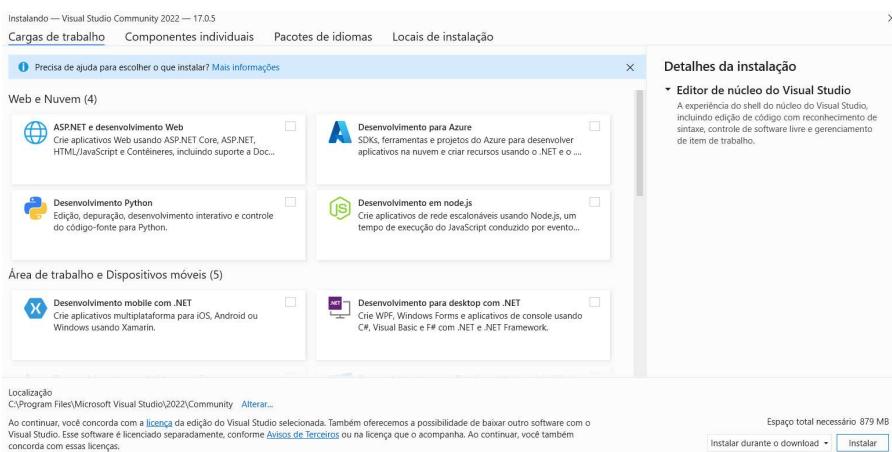
O .NET é *open source* e está sob a **.NET Foundation**, uma organização independente que promove o desenvolvimento aberto e a colaboração em torno do ecossistema .NET (MICROSOFT, c2022).

Instalando o Visual Studio

A Microsoft possui uma série de softwares voltados para programação, dentre eles há o **IDE Visual Studio**, o **Visual Studio para Mac** e o **Visual Studio Code**, que é mais voltado para aplicações web. No momento, vamos explanar o Visual Studio, que é a IDE de desenvolvimento para **.NET e C++ no Windows**.

Para fazer download do Visual Studio, basta acessar o site: <https://visualstudio.microsoft.com/pt-br/free-developer-offers/>. Depois de fazer o download, basta executar o instalador e rodar o aplicativo; em seguida, a tela inicial será mostrada, como na Figura 3.

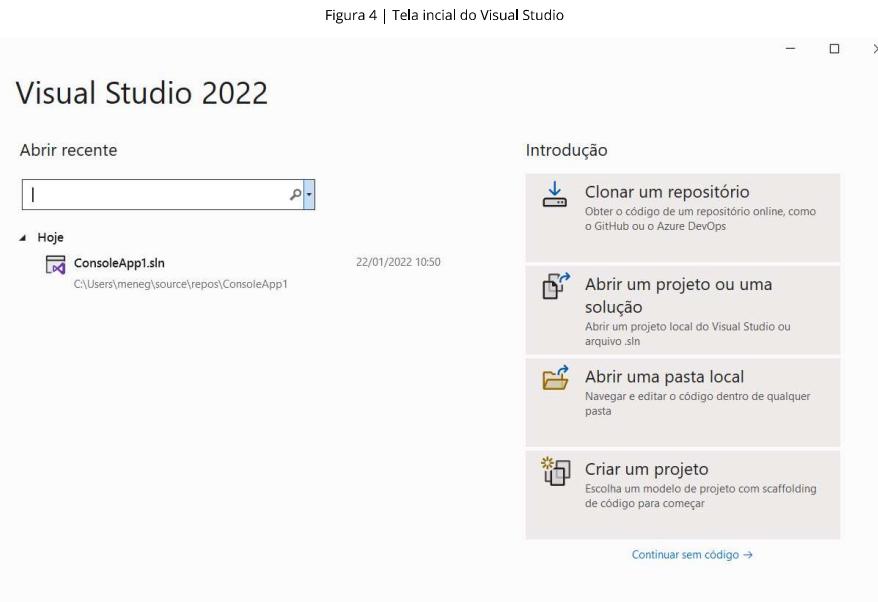
Figura 3 | Tela de instalação



Fonte: captura de tela de Visual Studio.

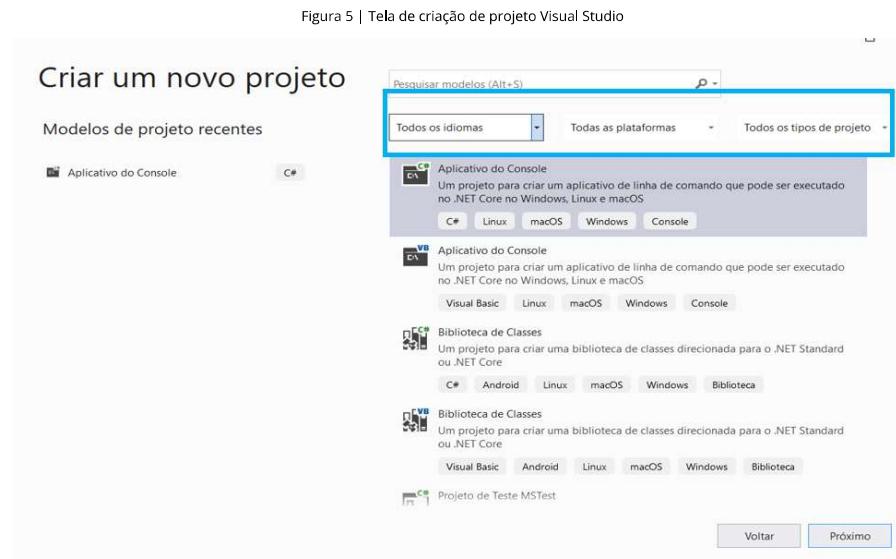
Na tela de instalação, é possível escolher qual linguagem você utilizará para programar. A partir disso, ele já prepara o ambiente, deixando-o pronto para programação. Nesse caso, a seleção deve ficar para o desenvolvimento para desktop com .NET. Caso deseje utilizar programação para outra linguagem ou para outros ambientes, como desenvolvimento *mobile* ou web, basta selecionar as fontes que estão presentes, e ele fará todo o processo de instalação automaticamente.

A tela de inicialização mostrará várias opções que podem ser selecionadas para utilizar o framework .NET, como mostrado na Figura 4.



Fonte: captura de tela de Visual Studio.

Nessa tela podemos ver que existem várias opções para se realizar a criação de um projeto ou para continuá-lo. Ao clicar em "Criar um projeto", ele abre um leque de opções, o que nos mostra a quantidade de opções que a plataforma traz.



Fonte: captura de tela adaptada de Visual Studio.

Na marcação evidenciada anteriormente, podemos identificar quais são as linguagens as plataformas e os tipos de projetos que podem ser criados. Claramente se vê quanto a ferramenta é poderosa para a construção de qualquer aplicativo. Dessa forma, é só fazer a escolha do aplicativo que a estrutura da ferramenta se encarrega de fazer os demais passos.

VIDEOAULA: .NET: CASES UTILIZANDO O VISUAL STUDIO

No vídeo referente ao bloco, será explanado sobre o **toolkit .NET**, sobre suas facilidades e também sobre as integrações que ela pode fazer com a construção de qualquer aplicativo. A **IDE Visual Studio** será também explanada de forma a mostrar as melhores ferramentas disponíveis, levando em consideração, ainda, a loja de complementos que ela possui.

MOBILE: CASES UTILIZANDO O ANDROID STUDIO

Desenvolvimento mobile

O desenvolvimento de aplicativos móveis é o processo de criação de software para smartphones e assistentes digitais, o que ocorre mais comumente para **Android** e **iOS**. O software pode ser pré-instalado no dispositivo, baixado de uma loja de aplicativos ou acessado por meio de um navegador da web móvel. As linguagens de programação e marcação usadas para esse tipo de desenvolvimento de software incluem **Java**, **Swift**, **C#** e **HTML5** (IBM CLOUD EDUCATION, 2020).

O desenvolvimento de aplicativos móveis está crescendo rapidamente. De varejo, telecomunicações e comércio eletrônico a seguros, saúde e governo, as organizações de todos os setores devem atender às expectativas dos usuários por meios convenientes e em tempo real de realizar transações e acessar informações. Hoje, os dispositivos móveis – e os aplicativos móveis que desbloqueiam seu valor – são a maneira mais popular de pessoas e empresas se conectarem à internet. Para permanecerem relevantes, responsivas e bem-sucedidas, as organizações precisam desenvolver os aplicativos móveis que seus clientes, parceiros e funcionários exigem (IBM CLOUD EDUCATION, 2020).

No entanto, o desenvolvimento de aplicativos móveis pode parecer assustador. Depois de selecionar a plataforma, ou plataformas, do sistema operacional, você precisa superar as limitações dos dispositivos móveis e conduzir seu aplicativo até os possíveis obstáculos de distribuição. Felizmente, seguindo algumas diretrizes básicas e práticas recomendadas, você pode otimizar sua jornada de desenvolvimento de aplicativos (IBM CLOUD EDUCATION, 2020).

Android Studio: instalação

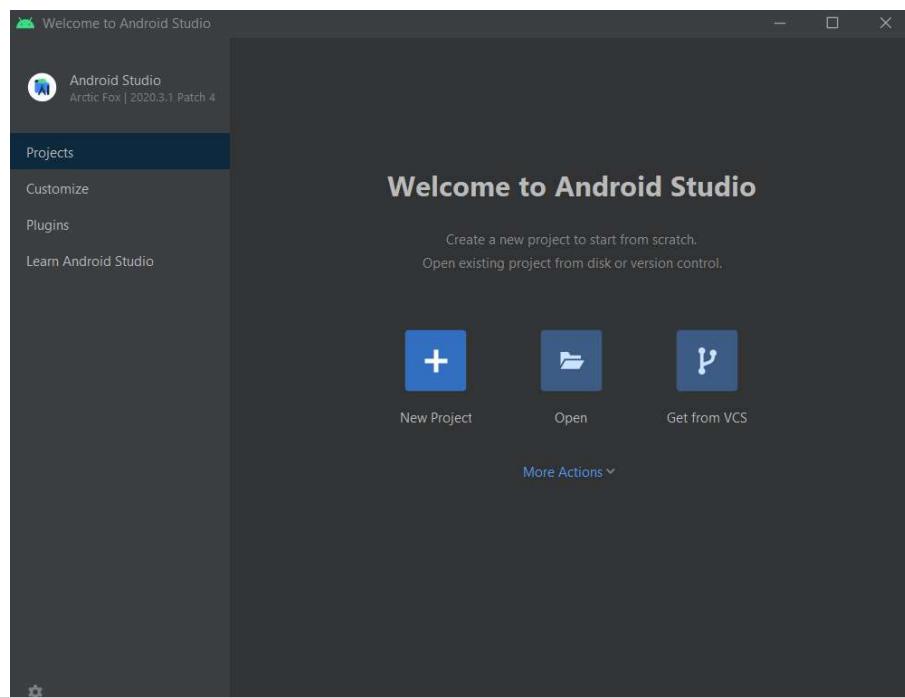
O **IDE Android Studio** é um ambiente de desenvolvimento integrado oficial para o desenvolvimento de aplicativos **Android**. Além do editor de código e das ferramentas de desenvolvimento avançadas, ele oferece ainda mais recursos para aumentar a produtividade na criação de aplicativos. Alguns exemplos de recursos são (DEVELOPER, 2021):

- Emulador com recursos.
- Ambiente unificado que possibilita o desenvolvimento para todos os dispositivos Android.
- Frameworks.

Para instalar o **Android Studio** é bem simples: basta fazer o download em:

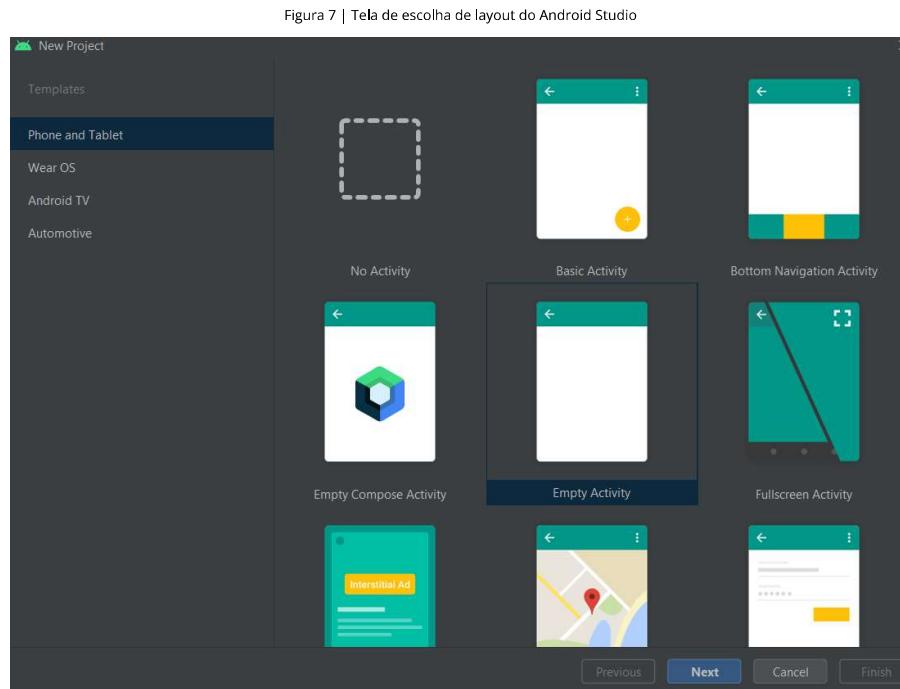
<https://developer.android.com/studio>. Após o download, execute e conclua as configurações padrões. A tela inicial será mostrada como na Figura 6.

Figura 6 | Tela inicial Android Studio



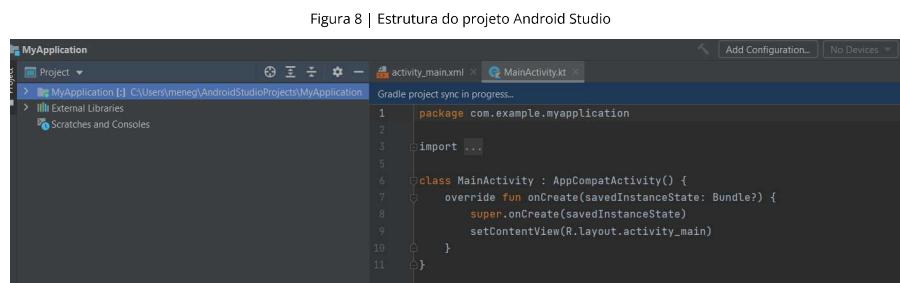
Fonte: captura de tela de Android Studio.

Após aberta a página inicial do **Android Studio**, basta criar um novo projeto. A partir disso, você será levado a uma janela para escolha de layout, na qual você poderá trabalhar.



Fonte: captura de tela de Android Studio.

Depois de escolher e avançar no layout, a IDE fará várias atualizações de pacotes e downloads necessários para trabalhar com a arquitetura escolhida.



Fonte: captura de tela de Android Studio.

A Figura 8 reflete como o ambiente ficará depois de executado o software. Nesse caso, pode-se iniciar a programação na linguagem escolhida. Vale lembrar que o Android Studio necessita de recursos de máquina, pois demanda um processamento razoável. Para Windows, por exemplo, vale a configuração prescrita no site, como (DEVELOPER, 2021):

- 8 GB de memória RAM ou mais.
- 8 GB de espaço de disco no mínimo.
- Resolução de tela de 1280 x 800 no mínimo.

Para outros tutoriais, o site do Android Studio pode ajudá-lo: <https://developer.android.com/studio>. Acesso em: 22 jan. 2022.

VIDEOAULA: MOBILE: CASES UTILIZANDO O ANDROID STUDIO

No vídeo referente ao bloco, serão mostradas algumas vantagens de desenvolvimento **mobile** e como o mercado está aquecido para esse nicho de mercado. Deverá mostrar também a instalação e a execução do **Android Studio** e algumas das suas principais ferramentas disponíveis.

ESTUDO DE CASO

Imagine que você trabalha em uma empresa de tecnologia e que seus superiores solicitaram ideias para construção de aplicativos de dispositivos móveis, tendo em vista que o aumento da demanda no mundo tecnológico está cada vez maior para aplicativos móveis e que a empresa deve evoluir para acompanhar o mercado.

Nesse sentido, é solicitado a você que encontre plataformas, linguagens de programação e ferramentas que podem ser implantadas nesse cenário, sendo que o objetivo é aumentar as opções de ferramentas para os clientes, tornando a empresa mais conhecida e com mais facilidades para trabalhar com o sistema corporativo via dispositivo móvel.

RESOLUÇÃO DO ESTUDO DE CASO

Caro aluno, para iniciar a análise desse caso, devemos fazer uma busca pelas melhores linguagens de mercado, ou seja, aquelas que estão em alta e que já se consolidaram. Nesse sentido, podemos pensar em linguagens de programação como **Java, C#, .NET, Python, C, Ruby** e tantas outras.

Por se tratar de uma solicitação para programação para dispositivos móveis, algumas dessas linguagens não se encaixam para criar aplicativos móveis. Portanto, teríamos algumas opções, como **Java, Swift, C#, Ruby**. No sentido de ferramentas e frameworks para tal propósito, todas elas oferecem inúmeras opções. Com isso, basta que você opte pela ferramenta que mais lhe agrada ou em que tenha mais conhecimento.

Resolução do Estudo de Caso

Para visualizar o objeto, acesse seu material digital.

6º Saiba mais

Para saber mais sobre desenvolvimento para dispositivos Android, você poderá utilizar a página oficial do Android Studio disponível em: <https://developer.android.com/>.

REFERÊNCIAS

7 minutos

Aula 1

AWS. AWS Documentation – Find users guides, developer guides, API references, tutorials, and more. **AWS**, [S. /], c2022. Disponível em: https://docs.aws.amazon.com/index.html?nc2=h_ql_doc_d0. Acesso em: 3 jan. 2022.

CHEBBI, A. Desenvolvimento de app: escolhendo a linguagem de programação. **Tecmundo**, São Paulo, 1 out. 2020. Disponível em: <https://www.tecmundo.com.br/software/204629-desenvolvimento-app-escolhendo-linguagem-programacao.htm>. Acesso em: 3 jan. 2022.

ELER, M. M. **Um método para o desenvolvimento de software baseado em componentes e aspectos**. 2006. 172 f. Dissertação (Mestrado em Ciência da Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2006. Disponível em: <https://pdfs.semanticscholar.org/df01/4a8fc39b229aba52e05c68c7f7405dc4ef13.pdf>. Acesso em: 1 jan. 2022.

FAYAD, M.; SCHMIDT, D. C. Object-oriented application frameworks. **Communications of the ACM**, [S. /], v. 40, n. 10, p. 32-38, 1997.

FIRESMITH D. G. Frameworks: the golden path to object Nirvana. **Journal of Object-Oriented Programming**, [S. /], v. 7, n. 8, 1994.

FUJIOKA, R. da C. **Uma arquitetura de referência para gerenciamento e reuso de ambientes virtuais tridimensionais**. 2015. 163 f. Tese (Doutorado em Ciência da Computação) – Centro de Informática, Universidade Federal de Pernambuco, Recife, 2015. Disponível em:
<https://attena.ufpe.br/bitstream/123456789/26271/1/TESE%20Rodrigo%20da%20Cruz%20Fujioka.pdf>. Acesso em: 1 jan. 2022.

GHIMIRE, D. Comparative study on Python web frameworks: Flask and Django. **Theseus**, [S. I.], 2020. Disponível em: <https://www.theseus.fi/handle/10024/339796>. Acesso em: 3 fev. 2022.

IBM CLOUD EDUCATION. **Application Programming Interface (API)**. IBM, [S. I.], 19 ago. 2020. Disponível em: <https://www.ibm.com/cloud/learn/api>. Acesso em: 3 fev. 2022.

KALUŽA, M.; KALANJ, M.; VUKELIĆ, B. A comparison of back-end frameworks for web application development. **Zbornik veleučilišta u rijeci**, [S. I.], v. 7, n. 1, p. 317-332, 2019.

MATTSSON, M. **Object-Oriented Frameworks**: a survey of methodological issues. Lund: Lund University, 1996. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.36.1424&rep=rep1&type=pdf>. Acesso em: 1 jan. 2022.

Aula 2

ANACONDA. **Data science technology for groundbreaking**. ANACONDA, [S. I.], c2022. Disponível em: <https://www.anaconda.com/>. Acesso em: 10 jan. 2022.

BRESSON, J. et al. **Platform. Eclipse Foundation**, [S. I.], 24 mar. 2016. Disponível em: <https://wiki.eclipse.org/Platform>. Acesso em: 10 jan. 2022.

CODE. Code editing. Redefined. **Visual Studio Code**, [S. I.], c2022. Disponível em: <https://code.visualstudio.com/>. Acesso em: 10 jan. 2022.

FLUTTER. Flutter documentation. **Flutter**, [S. I., s. d.]. Disponível em: <https://docs.flutter.dev/>. Acesso em: 30 dez. 2021.

IBM CLOUD EDUCATION. Data Science. IBM, [S. I.], 15 maio 2020. Disponível em: <https://www.ibm.com/cloud/learn/data-science-introduction>. Acesso em: 30 dez. 2021.

NETBEANS IDE. A maneira mais inteligente e rápida de programar. ORACLE, [S. I.], c2022. Disponível em: <https://www.oracle.com/br/tools/technologies/netbeans-ide.html>. Acesso em: 10 jan. 2022.

PROKOFYeva, N.; BOLTUNOVA, V. Analysis and practical application of PHP frameworks in development of web information systems. **Procedia Computer Science**, [S. I.], v. 104, p. 51-56, 2017.

WEB FRAMEWORKS. **Python Wiki**. [S. I.]: Python, 2021. Disponível em: <https://wiki.python.org/moin/WebFrameworks>. Acesso em: 30 dez. 2021.

REDHAT. **O que são frameworks Java? Red Hat**, [S. I.], 17 dez. 2019. Disponível em: <https://www.redhat.com/pt-br/topics/cloud-native-apps/what-is-a-Java-framework>. Acesso em: 10 jan. 2022.

ZUSE, H. **A framework of software measurement**. Berlin: New York: Walter de Gruyter, 2013.

Aula 3

CODE. Code editing. Redefined. **Visual Studio Code**, [S. I.], c2022a. Disponível em: <https://code.visualstudio.com/>. Acesso em: 10 jan. 2022.

CODE. **Getting Started. Visual Studio Code**, [S. I.], c2022b. Disponível em: <https://code.visualstudio.com/docs>. Acesso em: 21 jan. 2022.

CODE. PHP in Visual Studio Code. **Visual Studio Code**, [S. I.], c2022c. Disponível em: <https://code.visualstudio.com/docs/languages/php>. Acesso em: 21 jan. 2022.

ECLIPSE. Download Eclipse Technology that is right for you. **Eclipse Foundation**, [S. I.], 2021. Disponível em: <https://www.eclipse.org/downloads/>. Acesso em: 20 jan. 2022.

JAVA. **O que é a Tecnologia Java e porque preciso dela? Java**, [S. I., s. d.]. Disponível em: https://www.java.com/pt-BR/download/help/whatis_java.html. Acesso em: 20 jan. 2022.

JAVASCRIPT. **An Introduction to JavaScript**. **JavaScript Info**, [S. I.], 12 dez. 2021. Disponível em: <https://javascript.info/intro>. Acesso em: 20 jan. 2022.

JETBRAINS. WebStorm – **O IDE JavaScript mais inteligente.** JET BRAINS, [S. /], c2022. Disponível em: <https://www.jetbrains.com/pt-br/webstorm/>. Acesso em: 20 jan. 2022.

MDN. **A re-introduction to JavaScript.** MDN Web Docs – Mozilla, [S. /], 2022. Disponível em: https://developer.mozilla.org/en-US/docs/Web/JavaScript/A_re-introduction_to_JavaScript#overview, Acesso em: 20 jan. 2022.

NETBEANS. Apache NetBeans 12.6. **Apache NetBeans,** [S. /], 29 nov. 2021. Disponível em: <https://netbeans.apache.org/download/index.html>. Acesso em: 20 jan. 2022.

NIEDERAUER, J. **Desenvolvendo Websites com PHP:** Aprenda a criar Websites dinâmicos e interativos com PHP e bancos de dados. São Paulo: Novatec Editora, 2017.

ORACLE. Java Documentation. **Oracle Help Center,** [S. /, s. d.]. Disponível em: <https://docs.oracle.com/en/java/>. Acesso em: 21 jan. 2022.

PHP. **PHP,** [S. /], 2022. Disponível em: <https://www.php.net/>. Acesso em: 21 jan. 2022.

WINDOWS. PHP 8.1 (8.1.2). **PHP:** Hypertext Preprocessor. Disponível em: <https://windows.php.net/download#php-8.1>. Acesso em: 21 jan. 2022.

Aula 4

ANACONDA. **Your data science toolkit. Anaconda,** [S. /], c2022. Disponível em: <https://www.anaconda.com/products/individual>. Acesso em: 21 jan. 2022.

BAAREN, E. van. Install Python: Detailed Instructions for Window, Mac, and Linux. **Python Land,** [S. /], 8 jan. 2022. Disponível em: <https://python.land/installing-python>. Acesso em: 21 jan. 2022.

DEVELOPER. Android Studio. **Developers,** [S. /], 2021. Disponível em: <https://developer.android.com/studio>. Acesso em: 22 jan. 2022.

DEVELOPER. Apresentando os desenvolvedores Android. **Developers,** [S. /, s. d.]. Disponível em: <https://developer.android.com/>. Acesso em: 22 jan. 2022.

IBM CLOUD EDUCATION. **Mobile Application Development.** IBM, [S. /], 21 dez. 2020. Disponível em: <https://www.ibm.com/cloud/learn/mobile-application-development-explained>. Acesso em: 22 jan. 2022.

JETBRAINS. Baixar PyCharm. **JETBRAIN,** [S. /], 2022. Disponível em: <https://www.jetbrains.com/pt-br/pycharm/download/#section=windows>. Acesso em: 22 jan. 2022.

MICROSOFT. **What is .NET? Microsoft,** [S. /], c2022. Disponível em: <https://dotnet.microsoft.com/en-us/learn/dotnet/what-is-dotnet>. Acesso em: 22 jan. 2022.

VISUAL STUDIO. Tudo o que você precisa para criar ótimos aplicativos. Gratuito. **Visual Studio – Microsoft,** [S. /], c2022. Disponível em: <https://visualstudio.microsoft.com/pt-br/free-developer-offers/>. Acesso em: 22 jan. 2022.