

Mestrado Próprio

Inteligência Artificial na Programação



Mestrado Próprio Inteligência Artificial na Programação

- » Modalidade: online
- » Duração: 6 meses
- » Certificação: TECH Universidade Tecnológica
- » Acreditação: 90 ECTS
- » Horário: a tua scelta
- » Exames: online

Acesso ao site: www.techtute.com/pt/inteligencia-artificial/mestrado-proprio/mestrado-proprio-inteligencia-artificial-programacao

Índice

01

Apresentação

pág. 4

02

Objetivos

pág. 8

03

Competências

pág. 16

04

Direção do curso

pág. 20

05

Estrutura e conteúdo

pág. 24

06

Metodología de estudo

pág. 44

07

Certificação

pág. 50

01

Apresentação

Numa era caracterizada pelos avanços tecnológicos, a Inteligência Artificial (IA) surgiu como uma ferramenta fundamental para os especialistas em Programação. A sua importância reside na sua capacidade de mecanizar atividades árduas, tomar decisões com base em dados precisos e aprender com os padrões. Assim, a Aprendizagem Automática oferece valiosas técnicas para que os informáticos desenvolvam sistemas mais inteligentes. Estas vão desde algoritmos para dotar os programas de uma maior minúcia até ao desenvolvimento de sistemas autónomos, úteis para mudar a forma como os códigos são executados. Por este motivo, a TECH lança um curso que proporcionará aos estudantes os últimos avanços neste domínio. Tudo sob uma metodologia 100% online, adaptada à agenda de profissionais ocupados.



“

*Poderá conceber experiências de
utilizador personalizadas e intuitivas
através deste programa 100% online”*

A Inteligência Computacional ajuda as instituições a melhorar a produtividade no desenvolvimento de *software*. As suas ferramentas têm a capacidade de tratar dados não estruturados, aprender com experiências passadas e adaptar-se a mudanças em ambientes dinâmicos. Além disso, a IA pode prever potenciais problemas de aplicação antes de estes ocorrerem, permitindo que os profissionais tomem medidas preventivas para evitar problemas dispendiosos no futuro. Neste contexto, as principais empresas informáticas internacionais procuram ativamente recrutar especialistas em Arquitetura de Software para QA *Testing*.

Por conseguinte, a TECH implementa um programa inovador para que os programadores tirem o máximo partido da otimização e da gestão do desempenho nas ferramentas de IA. Concebido por especialistas de renome, o programa de estudos abordará os algoritmos de programação para desenvolver produtos com sistemas inteligentes. Adicionalmente, o programa abordará as extensões essenciais para o Visual Studio Code, o editor de código fonte mais utilizado atualmente. Além disso, os materiais de formação abordarão a integração da IA na gestão com bases de dados para detetar potenciais falhas e criar *test* unitários. Trata-se de um Mestrado Próprio que possui uma diversidade de conteúdos audiovisuais em múltiplos formatos e uma rede de simulações reais para aproximar o desenvolvimento do programa à realidade da prática informática.

A fim de atingir os objetivos de aprendizagem propostos, este programa é ministrado através de uma metodologia de ensino em online. Desta forma, os profissionais poderão conciliar perfeitamente o seu trabalho com os seus estudos. Além disso, o estudante beneficiará de um corpo docente de primeira classe e de materiais académicos multimédia de grande rigor pedagógico, tais como aulas magistrais, resumos interativos e exercícios práticos. O único requisito para aceder ao Campus Virtual é que os alunos tenham um dispositivo eletrónico com acesso à Internet, mesmo que seja o telemóvel.

Este **Mestrado Próprio em Inteligência Artificial na Programação** conta com o conteúdo educacional mais completo e atualizado do mercado. As suas principais características são:

- O desenvolvimento de casos práticos apresentados por especialistas em Inteligência Artificial em Programação
- Os conteúdos gráficos, esquemáticos e predominantemente práticos com que está concebido fornecem informações científicas e práticas sobre as disciplinas que são essenciais para a prática profissional
- Os exercícios práticos onde o processo de autoavaliação pode ser efetuado a fim de melhorar a aprendizagem
- O seu foco especial em metodologias inovadoras
- As aulas teóricas, perguntas ao especialista, fóruns de discussão sobre questões controversas e atividades de reflexão individual
- A disponibilidade de acesso aos conteúdos a partir de qualquer dispositivo fixo ou portátil com conexão à Internet



Obterá uma perspetiva holística sobre a forma como a Aprendizagem Automática afeta e melhora todas as fases do desenvolvimento de software”

“

Pretende aplicar Modelos Transformers para o processamento de linguagem natural na sua prática? Consegui-lo graças a este inovador programa”

O corpo docente do programa inclui profissionais do setor que trazem a experiência do seu trabalho para esta formação, bem como especialistas reconhecidos das principais sociedades e universidades de prestígio.

Os seus conteúdos multimédia, desenvolvidos com a mais recente tecnologia educativa, permitirão ao profissional uma aprendizagem situada e contextual, ou seja, um ambiente simulado que proporcionará uma formação imersiva programada para treinar em situações reais.

O design deste programa foca-se na Aprendizagem Baseada em Problemas, através da qual o profissional deverá tentar resolver as diferentes situações da atividade profissional que surgem ao longo do curso. Para tal, contará com a ajuda de um sistema inovador de vídeo interativo desenvolvido por especialistas reconhecidos.

Irá aprofundar o ciclo de vida do Testing, desde a criação de casos de teste até à deteção de bugs.

O Relearning permitir-lhe-á aprender com menos esforço e mais rendimento, envolvendo-o mais na sua especialização profissional.



02

Objetivos

Esta formação tornará os informáticos em especialistas em IA aplicada à Programação. Os estudantes adquirirão uma visão integral, combinando os conhecimentos mais atualizados com competências práticas que melhorarão a sua tomada de decisões. Ao mesmo tempo, os profissionais dominarão as ferramentas mais modernas para o desenvolvimento de *softwares* impulsionados pela Aprendizagem Automática. Desta forma, os alunos conceberão propostas quer para sítios Web quer para aplicações móveis com adaptabilidade. Será altamente especializado para responder às exigências atuais da indústria.





“

*Pretende especializar-se em Inteligência Artificial?
Com este programa, irá dominar a otimização
do processo de implantação e a integração de
Inteligência Artificial na computação em nuvem”*



Objetivos gerais

- ♦ Desenvolver competências para criar e gerir ambientes de desenvolvimento eficientes, garantindo uma base sólida para a implementação de projetos com IA
- ♦ Adquirir competências em matéria de planeamento, execução e automatização de testes de qualidade, incorporando ferramentas de IA para a deteção e correção de *bugs*
- ♦ Compreender e aplicar princípios de desempenho, escalabilidade e facilidade de manutenção na conceção de sistemas informáticos de grande escala
- ♦ Familiarize-se com os padrões de design mais importantes e aplicá-los eficazmente na arquitetura de software



Objetivos específicos

Módulo 1. Fundamentos da Inteligência Artificial

- ♦ Analisar a evolução histórica da Inteligência Artificial, desde o seu início até ao seu estado atual, identificando os principais marcos e desenvolvimentos-chave
- ♦ Compreender o funcionamento das redes neuronais e a sua aplicação em modelos de aprendizagem em Inteligência Artificial
- ♦ Estudar os princípios e aplicações dos algoritmos genéticos, analisando a sua utilidade na resolução de problemas complexos
- ♦ Analisar a importância dos thesauri, vocabulários e taxonomias na estruturação e processamento de dados para sistemas de IA
- ♦ Explorar o conceito de web semântica e a sua influência na organização e compreensão da informação em ambientes digitais

Módulo 2. Tipos e ciclo de vida do dado

- ♦ Compreender os conceitos fundamentais da estatística e a sua aplicação na análise de dados
- ♦ Identificar e classificar os diferentes tipos de dados estatísticos, desde os quantitativos aos qualitativos
- ♦ Analisar o ciclo de vida dos dados, desde a sua geração até à sua eliminação, identificando as principais etapas
- ♦ Explorar as fases iniciais do ciclo de vida dos dados, destacando a importância do planeamento e da estrutura dos dados
- ♦ Estudar os processos de recolha de dados, incluindo a metodologia, as ferramentas e os canais de recolha
- ♦ Explorar o conceito de *Datawarehouse* (Armazenamento de Dados), com ênfase nos seus elementos constituintes e no seu design
- ♦ Analisar os aspetos regulamentares relacionados com a gestão de dados, cumprindo as normas de privacidade e segurança, bem como as boas práticas

Módulo 3. O dado na Inteligência Artificial

- ♦ Dominar os fundamentos da ciência dos dados, abrangendo ferramentas, tipos e fontes de análise de informações
- ♦ Explorar o processo de transformação de dados em informação utilizando técnicas de mineração e visualização de dados
- ♦ Estudar a estrutura e características dos *datasets*, compreendendo a sua importância na preparação e utilização de dados para modelos de Inteligência Artificial
- ♦ Analisar modelos supervisionados e não supervisionados, incluindo métodos e classificação
- ♦ Utilizar ferramentas específicas e boas práticas no tratamento e processamento de dados, garantindo eficiência e qualidade na implementação de Inteligência Artificial

Módulo 4. Mineração de dados. Seleção, pré-processamento e transformação

- ♦ Dominar técnicas de inferência estatística para compreender e aplicar métodos estatísticos na mineração de dados
- ♦ Realizar análises exploratórias pormenorizadas de conjuntos de dados para identificar padrões, anomalias e tendências relevantes
- ♦ Desenvolver competências para a preparação de dados, incluindo a sua limpeza, integração e formatação para utilização na mineração de dados
- ♦ Implementar estratégias eficazes para tratar valores em falta em conjuntos de dados, aplicando métodos de imputação ou eliminação sensíveis ao contexto
- ♦ Identificar e atenuar o ruído nos dados, utilizando técnicas de filtragem e suavização para melhorar a qualidade do conjunto de dados
- ♦ Abordar o pré-processamento de dados em ambientes *Big Data*

Módulo 5. Algoritmo e complexidade na Inteligência Artificial

- ♦ Introduzir estratégias de conceção de algoritmos, proporcionando uma compreensão sólida das abordagens fundamentais para a resolução de problemas
- ♦ Analisar a eficiência e a complexidade dos algoritmos, aplicando técnicas de análise para avaliar o desempenho em termos de tempo e espaço
- ♦ Estudar e aplicar algoritmos de ordenação, compreendendo o seu desempenho e comparando a sua eficiência em diferentes contextos
- ♦ Explorar algoritmos baseados em árvores, compreendendo a sua estrutura e aplicações
- ♦ Investigar algoritmos com *Heaps*, analisando a sua implementação e utilidade na manipulação eficiente de dados
- ♦ Analisar algoritmos baseados em grafos, explorando a sua aplicação na representação e resolução de problemas que envolvam relações complexas
- ♦ Estudar algoritmos *Greedy*, compreendendo a sua lógica e aplicações na resolução de problemas de otimização
- ♦ Investigar e aplicar a técnica de *backtracking* na resolução sistemática de problemas, analisando a sua eficácia numa variedade de cenários

Módulo 6. Sistemas inteligentes

- ♦ Explorar a teoria dos agentes, compreendendo os conceitos fundamentais do seu funcionamento e a sua aplicação na Inteligência Artificial e na engenharia de Software
- ♦ Estudar a representação do conhecimento, incluindo a análise de ontologias e a sua aplicação na organização de informação estruturada
- ♦ Analisar o conceito de web semântica e o seu impacto na organização e recuperação de informação em ambientes digitais

- ♦ Avaliar e comparar diferentes representações do conhecimento, integrando-as para melhorar a eficiência e a precisão dos sistemas inteligentes
- ♦ Estudar raciocinadores semânticos, sistemas baseados no conhecimento e sistemas periciais, compreendendo a sua funcionalidade e aplicações na tomada de decisões inteligentes

Módulo 7. Aprendizagem automática e mineração de dados

- ♦ Introduzir processos de descoberta de conhecimentos e os conceitos fundamentais da aprendizagem automática
- ♦ Estudar árvores de decisão como modelos de aprendizagem supervisionada, compreendendo a sua estrutura e aplicações
- ♦ Avaliar classificadores utilizando técnicas específicas para medir o seu desempenho e exatidão na classificação de dados
- ♦ Estudar as redes neuronais, compreendendo o seu funcionamento e arquitetura para resolver problemas complexos de aprendizagem automática
- ♦ Explorar os métodos bayesianos e a sua aplicação na aprendizagem automática, incluindo redes bayesianas e classificadores bayesianos
- ♦ Analisar modelos de regressão e de resposta contínua para prever valores numéricos a partir de dados
- ♦ Estudar técnicas de *clustering* para identificar padrões e estruturas em conjuntos de dados não rotulados
- ♦ Explorar a extração de texto e o processamento de linguagem natural (PLN), compreendendo como as técnicas de aprendizagem automática são aplicadas para analisar e compreender texto

Módulo 8. As redes neuronais, a base da *Deep Learning*

- ♦ Dominar os fundamentos da Aprendizagem Profunda, compreendendo o seu papel essencial na *Deep Learning*
- ♦ Explorar as operações fundamentais nas redes neuronais e compreender a sua aplicação na construção de modelos
- ♦ Analisar as diferentes camadas utilizadas nas redes neuronais e aprender a seleccioná-las adequadamente
- ♦ Compreender a ligação eficaz de camadas e operações para conceber arquiteturas de redes neuronais complexas e eficientes
- ♦ Utilizar treinadores e otimizadores para ajustar e melhorar o desempenho das redes neuronais
- ♦ Explorar a ligação entre neurónios biológicos e artificiais para uma compreensão mais profunda da conceção de modelos
- ♦ Afinar hiperparâmetros para o *Fine Tuning* de redes neuronais, melhorando o seu desempenho em tarefas específicas

Módulo 9. Treino de redes neuronais profundas

- ♦ Resolver problemas relacionados com gradientes na formação de redes neuronais profundas
- ♦ Explorar e aplicar diferentes otimizadores para melhorar a eficiência e a convergência dos modelos
- ♦ Programar a taxa de aprendizagem para ajustar dinamicamente a velocidade de convergência do modelo
- ♦ Compreender e abordar o sobreajuste através de estratégias específicas durante o treino
- ♦ Aplicar diretrizes práticas para garantir o treino eficiente e eficaz de redes neuronais profundas
- ♦ Implementar a *Transfer Learning* como uma técnica avançada para melhorar o desempenho do modelo em tarefas específicas

- Explorar e aplicar técnicas de *Data Augmentation* para enriquecer conjuntos de dados e melhorar a generalização do modelo
- Desenvolver aplicações práticas utilizando a *Transfer Learning* para resolver problemas do mundo real
- Compreender e aplicar técnicas de regularização para melhorar a generalização e evitar o sobreajuste em redes neurais profundas

Módulo 10. Personalização de modelos e treino com *TensorFlow*

- Dominar os fundamentos do *TensorFlow* e a sua integração com o NumPy para um tratamento e computação eficientes dos dados
- Personalizar modelos e algoritmos de treino utilizando as capacidades avançadas do *TensorFlow*
- Explorar a API *tfddata* para gerir e manipular eficientemente conjuntos de dados
- Implementar o formato *TFRecord* para armazenar e aceder a grandes conjuntos de dados *TensorFlow*
- Utilizar camadas de pré-processamento do Keras para facilitar a construção de modelos personalizados
- Explorar o projeto *TensorFlow Datasets* para acessar conjuntos de dados predefinidos e melhorar a eficiência do desenvolvimento
- Desenvolver uma aplicação de *Deep Learning* com *TensorFlow*, integrando os conhecimentos adquiridos no módulo
- Aplicar de forma prática todos os conceitos aprendidos na construção e treino de modelos personalizados com *TensorFlow* em situações do mundo real

Módulo 11. *Deep Computer Vision* com Redes Neurais Convolucionais

- Compreender a arquitetura do córtex visual e a sua relevância para a *Deep Computer Vision*
- Explorar e aplicar camadas convolucionais para extrair características-chave de imagens
- Implementar camadas de agrupamento e sua utilização em modelos de *Deep Computer Vision* com o Keras

- Analisar várias arquiteturas de Redes Neurais Convolucionais (CNN) e a sua aplicabilidade em diferentes contextos
- Desenvolver e implementar uma CNN ResNet utilizando a biblioteca Keras para melhorar a eficiência e o desempenho do modelo
- Utilizar modelos Keras pré-treinados para tirar partido da aprendizagem por transferência para tarefas específicas
- Aplicar técnicas de classificação e localização em ambientes de *Deep Computer Vision*
- Explorar estratégias de deteção e seguimento de objetos utilizando Redes Neurais Convolucionais
- Implementar técnicas de segmentação semântica para compreender e classificar objetos em imagens de forma detalhada

Módulo 12. Processamento de linguagem natural (PLN) com Redes Neurais Recorrentes (RNN) e Atenção

- Desenvolver competências na geração de textos utilizando Redes Neurais Recorrentes (RNN)
- Aplicar RNN na classificação de opiniões para análise de sentimentos em textos
- Compreender e aplicar mecanismos de atenção em modelos de processamento de linguagem natural
- Analisar e utilizar modelos *Transformers* em tarefas específicas de PNL
- Explorando a aplicação de modelos *Transformers* no contexto do processamento de imagens e da visão computacional
- Familiarizar-se com a biblioteca *Transformers* de *Hugging Face* para a implementação eficiente de modelos avançados
- Comparar diferentes bibliotecas de *Transformers* para avaliar a sua adequação a tarefas específicas
- Desenvolver uma aplicação prática de PLN que integre RNN e mecanismos de atenção para resolver problemas do mundo real

Módulo 13. Autoencoders, GANs, e modelos de difusão

- ♦ Desenvolver representações de dados eficientes utilizando *Autoencoders*, *GANs* e Modelos de Difusão
- ♦ Realizar PCA utilizando um codificador automático linear incompleto para otimizar a representação dos dados
- ♦ Implementar e compreender o funcionamento de codificadores automáticos empilhados
- ♦ Explorar e aplicar codificadores automáticos convolucionais para representações visuais eficientes de dados
- ♦ Analisar e aplicar a eficácia dos codificadores automáticos esparsos na representação de dados
- ♦ Gerar imagens de moda a partir do conjunto de dados MNIST utilizando *Autoencoders*
- ♦ Compreender o conceito de Redes Generativas Antagônicas (*GANs*) e Modelos de Difusão
- ♦ Implementar e comparar o desempenho de modelos de difusão e *GANs* na geração de dados

Módulo 14. Computação bioinspirada

- ♦ Introduzir os conceitos fundamentais da computação bioinspirada
- ♦ Explorar os algoritmos de adaptação social como uma abordagem fundamental na computação bioinspirada
- ♦ Analisar estratégias de exploração do espaço em algoritmos genéticos
- ♦ Examinar modelos de computação evolutiva no contexto da otimização
- ♦ Continuar a análise pormenorizada de modelos de computação evolutiva
- ♦ Aplicar a programação evolutiva a problemas de aprendizagem específicos
- ♦ Abordar a complexidade de problemas multi-objetivo no âmbito da computação bioinspirada
- ♦ Explorar a aplicação de redes neuronais no domínio da computação bioinspirada
- ♦ Aprofundar a implementação e a utilidade das redes neuronais na computação bioinspirada

Módulo 15. Inteligência Artificial Estratégias e aplicações

- ♦ Desenvolver estratégias para a implementação da inteligência artificial nos serviços financeiros
- ♦ Analisar as implicações da inteligência artificial na prestação de serviços de saúde
- ♦ Identificar e avaliar os riscos associados à utilização da inteligência artificial no setor da saúde
- ♦ Avaliar os riscos potenciais associados à utilização da IA na indústria
- ♦ Aplicar técnicas de inteligência artificial na indústria para melhorar a produtividade
- ♦ Conceber soluções de inteligência artificial para otimizar os processos na administração pública
- ♦ Avaliar a aplicação de tecnologias de IA no setor da educação
- ♦ Aplicar técnicas de inteligência artificial na silvicultura e na agricultura para melhorar a produtividade
- ♦ Melhorar os processos de recursos humanos através da utilização estratégica da inteligência artificial

Módulo 16. Melhorar a produtividade no desenvolvimento de software com IA

- ♦ Aprofundar a implementação de extensões imprescindíveis de IA no Visual Studio Code para melhorar a produtividade e facilitar o desenvolvimento do *software*
- ♦ Obter uma sólida compreensão dos conceitos básicos de IA e da sua aplicação no desenvolvimento do *software*, incluindo algoritmos de aprendizagem automática, processamento de linguagem natural, redes neuronais, etc
- ♦ Dominar a configuração de ambientes de desenvolvimento otimizados, garantindo que os alunos possam criar ambientes propícios para projetos de IA
- ♦ Aplicar técnicas específicas utilizando o ChatGPT para identificação e correção automática de possíveis melhorias no código, incentivando práticas de programação mais eficientes
- ♦ Promover a colaboração entre diferentes profissionais de programação (desde programadores à engenheiros de dados e designers de experiência do utilizador) para desenvolver soluções do *software* com IA eficazes e éticas

Módulo 17. Arquitetura do software para QA Testing

- ♦ Desenvolver competências na conceção de planos de teste sólidos, abrangendo diferentes tipos de *testing* e garantindo a qualidade do software
- ♦ Reconhecer e analisar diferentes tipos de estruturas de *software*, como a monolítica, a de microsserviços ou a orientada para os serviços
- ♦ Obter uma visão global dos princípios e técnicas para a conceção de sistemas informáticos escaláveis e capazes de tratar grandes volumes de dados
- ♦ Aplicar conhecimentos avançados na implementação de estruturas de dados impulsionadas pela IA para otimizar o desempenho e a eficiência do *software*
- ♦ Desenvolver práticas de desenvolvimento seguras, centradas na prevenção de vulnerabilidades para garantir a segurança do *software* a nível arquitetónico

Módulo 18. Projetos Web com IA

- ♦ Desenvolver competências integrais para a implementação de projetos web, desde o design do *frontend* até a otimização do *backend*, com a inclusão de elementos de IA
- ♦ Otimizar o processo de implantação do sítio web, incorporando técnicas e ferramentas para melhorar a rapidez e a eficiência
- ♦ Integrar a IA na computação em nuvem, permitindo aos estudantes criar projetos web altamente escaláveis e eficientes
- ♦ Adquirir a capacidade de identificar problemas e oportunidades específicos em projetos web em que a IA pode ser aplicada eficazmente, como o processamento de texto, a personalização, a recomendação de conteúdos, etc
- ♦ Incentivar os estudantes a manterem-se a par das últimas tendências e desenvolvimentos em IA para uma aplicação adequada em projetos web

Módulo 19. Aplicações móveis com IA

- ♦ Aplicar conceitos avançados de *clean architecture*, *datasources* e *repositories* para garantir uma estrutura robusta e modular para aplicações móveis com IA
- ♦ Desenvolver competências para conceber ecrãs interativos, ícones e recursos gráficos utilizando a IA para melhorar a experiência do utilizador em aplicações móveis
- ♦ Aprofundar a configuração do ambiente de trabalho para as aplicações móveis e utilizar *Github Copilot* para agilizar o processo de desenvolvimento
- ♦ Otimizar as aplicações móveis com IA para um desempenho eficiente, tendo em conta a gestão de recursos e a utilização de dados
- ♦ Efetuar testes de qualidade em aplicações móveis com IA, permitindo aos alunos identificar problemas e depurar bugs

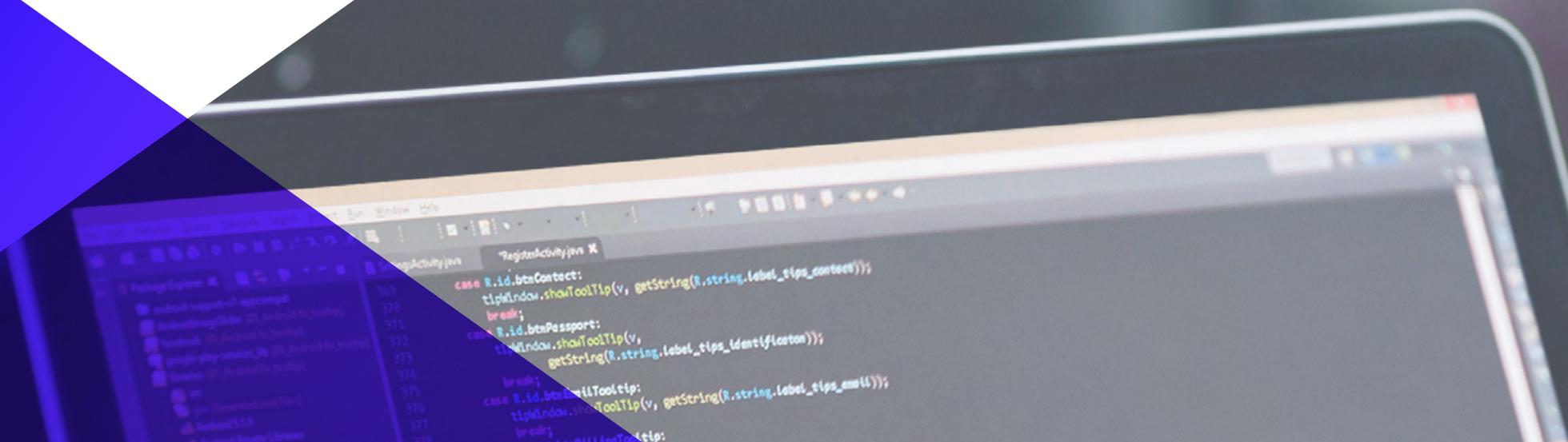
Módulo 20. IA para QA Testing

- ♦ Dominar princípios e técnicas de design de sistemas informáticos que sejam escaláveis e capazes de manusear grandes volumes de dados
- ♦ Aplicar conhecimentos avançados na implementação de estruturas de dados impulsionadas pela IA para otimizar o desempenho e a eficiência do *software*
- ♦ Compreender e aplicar práticas de desenvolvimento seguras, centradas em evitar vulnerabilidades como a injeção, para garantir a segurança do *software* a nível arquitetónico
- ♦ Gerar testes automatizados, especialmente em ambientes web e móveis, integrando ferramentas de IA para melhorar a eficiência do processo
- ♦ Utilizar ferramentas avançadas de controlo de qualidade com base em IA para uma deteção mais eficiente de *bugs* e a melhoria contínua do *software*

03

Competências

Ao longo de 12 meses de aprendizagem, os estudantes ficarão equipados com um conjunto de competências que elevarão os seus horizontes profissionais na indústria informática. Assim, os especialistas poderão criar e implementar softwares avançados e conceber propostas de IA em grande escala. Neste contexto, os profissionais abordarão a questão da produtividade no desenvolvimento, ao mesmo tempo que aplicam as melhores práticas de QA Testing. Este Mestrado Próprio garante que os informáticos são capazes de enfrentar com sucesso os desafios do mundo real e destacar-se significativamente numa área que está a avançar a passos largos.



“

Ficará altamente qualificado para implementar algoritmos de Aprendizagem Automática nos seus projectos web e aplicações móveis”

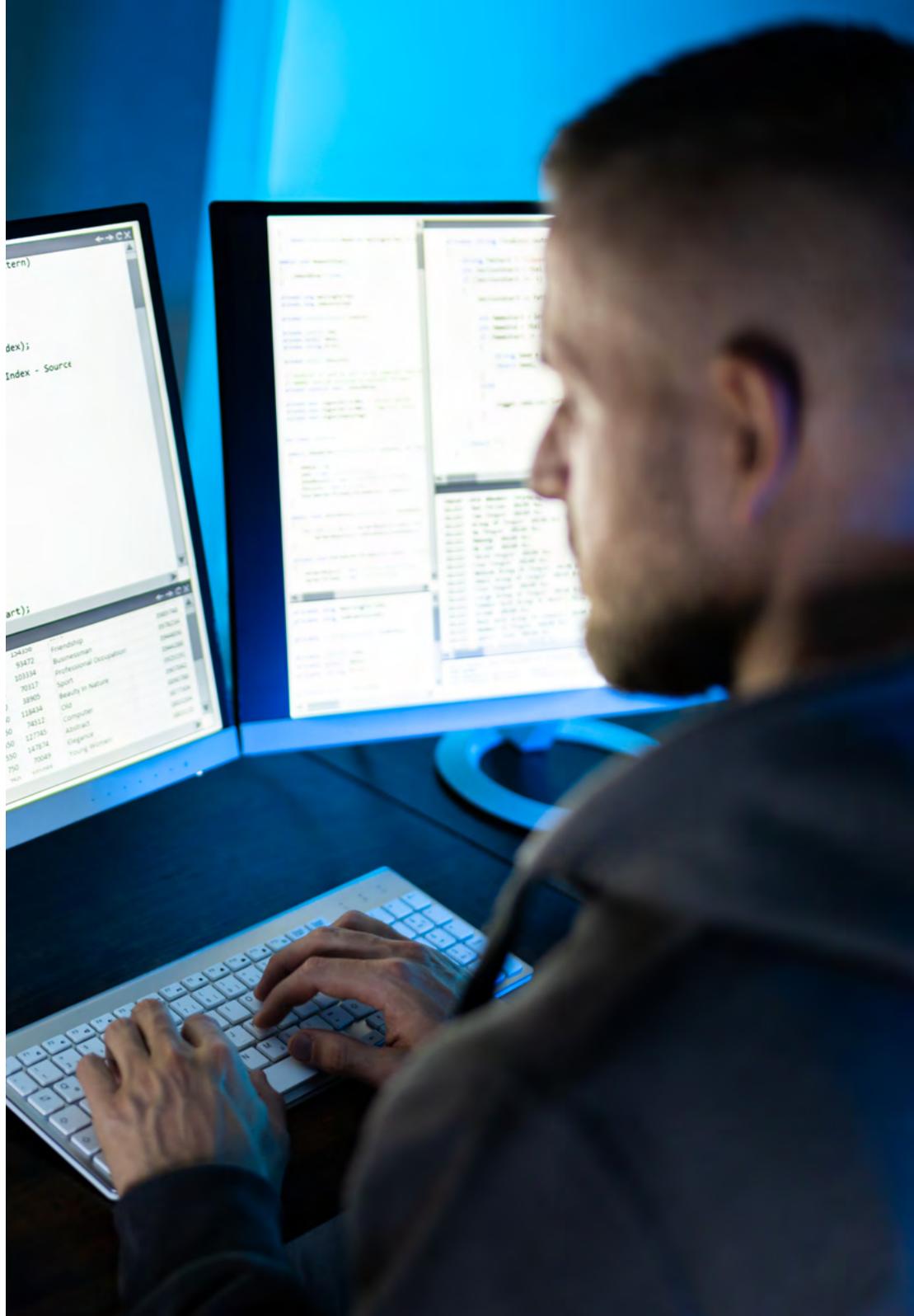


Competências gerais

- ♦ Aplicar extensões de IA em Visual Studio Code e técnicas de design *no-code* para aumentar a eficiência no desenvolvimento do software
- ♦ Utilizar o ChatGPT para otimizar e melhorar a qualidade do código, aplicando práticas de programação avançadas
- ♦ Implementar projetos web, desde a criação de *workspaces* até o despliegue, integrando IA, tanto em el *frontend* como no *backend*
- ♦ Desenvolver aplicações móveis alimentadas por IA, desde a configuração do ambiente até à criação de funções avançadas e à gestão de ativos gráficos
- ♦ Aplicar conceitos avançados de armazenamento e estruturas de dados impulsionadas por IA para melhorar a eficiência e a escalabilidade dos sistemas
- ♦ Incluir práticas de desenvolvimento seguras, evitando vulnerabilidades como a injeção, para garantir a integridade e a segurança do software desenvolvido



Dominarás as tecnologias do futuro com esta certificação exclusiva 100% online. Só com a TECH!





Competências específicas

- ♦ Aplicar técnicas e estratégias de IA para melhorar a eficiência no setor do *retalho*
- ♦ Implementar técnicas de redução de ruído utilizando codificadores automáticos
- ♦ Criar eficazmente conjuntos de dados de treino para tarefas de processamento de linguagem natural (NLP)
- ♦ Executar camadas de agrupamento e a sua utilização em modelos de *Deep Computer Vision* com o Keras
- ♦ Utilizar funções e gráficos *TensorFlow* para otimizar o desempenho de modelos personalizados
- ♦ Melhorar o desenvolvimento e a aplicação de *chatbots* e assistentes virtuais, compreendendo o seu funcionamento e potenciais aplicações
- ♦ Dominar a reutilização de camadas pré-treinadas para otimizar e acelerar o processo de treino
- ♦ Construir a primeira rede neural, aplicando na prática os conceitos aprendidos
- ♦ Ativar o Perceptron Multicamadas (MLP) usando a biblioteca Keras
- ♦ Aplicar técnicas de exploração e pré-processamento de dados, identificando e preparando os dados para uma utilização efectiva em modelos de aprendizagem automática
- ♦ Investigar linguagens e software para a criação de ontologias, utilizando ferramentas específicas para o desenvolvimento de modelos semânticos
- ♦ Desenvolver técnicas de limpeza de dados para garantir a qualidade e a exatidão da informação utilizada nas análises posteriores
- ♦ Dominar a configuração de ambientes de desenvolvimento otimizados, garantindo que os alunos possam criar ambientes propícios para projetos de IA
- ♦ Aplicar técnicas específicas, utilizando o ChatGPT para identificação e correção automática de possíveis melhorias no código, incentivando práticas de programação mais eficientes
- ♦ Criar testes automatizados, especialmente em ambientes web e móveis, integrando ferramentas de IA para melhorar a eficiência do processo
- ♦ Utilizar ferramentas avançadas de QA com base em IA para uma deteção mais eficiente de bugs e a melhoria contínua do software
- ♦ Integrar a IA na computação em nuvem, permitindo dos estudantes à criar projetos web altamente escaláveis e eficientes
- ♦ Configuração do ambiente de trabalho para as aplicações móveis e Utilizar Github Copilot para agilizar o processo de desenvolvimento

04

Direção do curso

Fiel ao seu compromisso de oferecer um ensino da mais alta qualidade, a TECH dispõe de um corpo docente do mais alto nível. Por esta razão, os especialistas que compõem este curso destacam-se pelo seu conhecimento profundo da IA em Programação, bem como pelos seus muitos anos de experiência profissional. Graças a estes aspetos, esta via académica oferece aos estudantes as melhores ferramentas e estratégias tecnológicas, não só para alargarem os seus conhecimentos, mas também para adquirirem as competências necessárias para aperfeiçoarem os seus procedimentos informáticos. Assim, os estudantes têm as garantias necessárias para se especializarem num sector digital que oferece numerosas oportunidades.



“

Os maiores especialistas em Aprendizagem Automática em Programação juntaram-se para partilhar consigo todo o seu conhecimento neste sector”

Direção



Dr. Peralta Martín-Palomino, Arturo

- ♦ CEO e CTO, Prometeus Global Solutions
- ♦ CTO em Korporate Technologies
- ♦ CTO em AI Shepherds GmbH
- ♦ Consultor e Assessor Empresarial Estratégico na Alliance Medical
- ♦ Diretor de Design e Desenvolvimento na DocPath
- ♦ Doutoramento em Engenharia Informática pela Universidade de Castilla-La Mancha
- ♦ Doutoramento em Economia, Empresas e Finanças pela Universidade Camilo José Cela
- ♦ Doutoramento em Psicologia pela Universidade de Castilla-La Mancha
- ♦ Mestrado em Executive MBA pela Universidade Isabel I
- ♦ Mestrado em Gestão Comercial e de Marketing pela Universidade Isabel I
- ♦ Mestrado Especialista em Big Data pela Formação Hadoop
- ♦ Mestrado em Tecnologias Avançadas de Informação da Universidade de Castilla-La Mancha
- ♦ Membro de: Grupo de Investigação SMILE



Sr. Castellanos Herreros, Ricardo

- ♦ *Chief Technology Officer* em OWQLO
- ♦ Especialista em Engenharia Informática de Sistemas e *Machine Learning Engineer*
- ♦ Consultor Técnico *Freelance*
- ♦ Desenvolvedor de Aplicações Móveis para eDreams, Fnac, Air Europa, Bankia, Cetelem, Banco Santander, Santillana, Groupon e Grupo Planeta
- ♦ Desenvolvedor de páginas Web para o Openbank e o Banco Santander
- ♦ Engenheiro Técnico em Informática de Sistemas pela Universidade de Castilla-La Mancha

05

Estrutura e conteúdo

Este Mestrado Próprio proporcionará aos estudantes uma abordagem holística, que lhes dará uma vantagem significativa no desenvolvimento informático e os dotará de competências específicas. Para tal, a formação irá desde a preparação do ambiente de desenvolvimento até à otimização do *software* y la implementación de IA en proyectos reales. O programa de estudos aprofundará aspectos como a conceção de *no-code* de interfaces, a utilização do ChatGPT para otimizar códigos ou a aplicação da Aprendizagem Automática em QA *Testing*. Desta forma, os alunos formados implementarão soluções inovadoras de forma eficaz em várias aplicações, tais como projectos web e móveis.



“

Atualize os seus conhecimentos sobre Inteligência Artificial em Programação através de conteúdos multimédia inovadores”

Módulo 1. Fundamentos da Inteligência Artificial

- 1.1. História da Inteligência Artificial
 - 1.1.1. Quando se começa a falar de inteligência artificial?
 - 1.1.2. Referências no cinema
 - 1.1.3. Importância da inteligência artificial
 - 1.1.4. Tecnologias que viabilizam e apoiam a inteligência artificial
- 1.2. Inteligência Artificial em jogos
 - 1.2.1. Teoria dos jogos
 - 1.2.2. *Minimax* e poda Alfa-Beta
 - 1.2.3. Simulação: Monte Carlo
- 1.3. Redes neuronais
 - 1.3.1. Fundamentos teológicos
 - 1.3.2. Modelo computacional
 - 1.3.3. Redes neuronais supervisionadas e não supervisionadas
 - 1.3.4. Perceptron simples
 - 1.3.5. Perceptron multicamadas
- 1.4. Algoritmos genéticos
 - 1.4.1. História
 - 1.4.2. Base biológica
 - 1.4.3. Codificação de problemas
 - 1.4.4. Criação da população inicial
 - 1.4.5. Algoritmo principal e operadores genéticos
 - 1.4.6. Avaliação dos indivíduos: Fitness
- 1.5. Tesouros, vocabulários, taxonomias
 - 1.5.1. Vocabulários
 - 1.5.2. Taxonomias
 - 1.5.3. Tesauro
 - 1.5.4. Ontologias
 - 1.5.5. Representação do conhecimento: web semântica
- 1.6. Web semântica
 - 1.6.1. Especificações: RDF, RDFS e OWL
 - 1.6.2. Inferência/razoabilidade
 - 1.6.3. *Linked Data*

- 1.7. Sistemas periciais e DSS
 - 1.7.1. Sistemas periciais
 - 1.7.2. Sistema de apoio à decisão
- 1.8. *Chatbots* e assistentes virtuais
 - 1.8.1. Tipos de assistentes: assistentes de voz e texto
 - 1.8.2. Partes fundamentais para o desenvolvimento de um assistente: *Intenções*, entidades e fluxo de diálogo
 - 1.8.3. Integração: Web, *Slack*, Whatsapp, Facebook
 - 1.8.4. Ferramentas para o desenvolvimento dos assistentes: *Dialog Flow*, *Watson Assistant*
- 1.9. Estratégia de implementação de IA
- 1.10. Futuro da inteligência artificial
 - 1.10.1. Compreendemos como detetar as emoções através de algoritmos
 - 1.10.2. Criação de uma personalidade: linguagem, expressões e conteúdo
 - 1.10.3. Tendências da Inteligência Artificial
 - 1.10.4. Reflexão

Módulo 2. Tipos e Ciclo de Vida do Dado

- 2.1. A Estatística
 - 2.1.1. Estatística: estatística descritiva, inferências estatísticas
 - 2.1.2. População, amostra indivíduo
 - 2.1.3. Variáveis: Definição de medição
- 2.2. Tipos de dados estatísticos
 - 2.2.1. De acordo com o tipo
 - 2.2.1.1. Quantitativos: dados contínuos e dados discretos
 - 2.2.1.2. Qualitativo: dados binomiais, dados nominais, dados ordinais
 - 2.2.2. De acordo com a sua forma
 - 2.2.2.1. Numérico
 - 2.2.2.2. Texto
 - 2.2.2.3. Lógico
 - 2.2.3. De acordo com a sua fonte
 - 2.2.3.1. Primários
 - 2.2.3.2. Secundários

- 2.3. Ciclo de vida dos dados
 - 2.3.1. Etapas do ciclo
 - 2.3.2. Marcos do ciclo
 - 2.3.3. Princípios FAIR
- 2.4. Etapas iniciais do ciclo
 - 2.4.1. Definição de metas
 - 2.4.2. Determinação de recursos necessários
 - 2.4.3. Diagrama de Gantt
 - 2.4.4. Estrutura de dados
- 2.5. Recolha de dados
 - 2.5.1. Metodologia de recolha
 - 2.5.2. Ferramentas de recolha
 - 2.5.3. Canais de recolha
- 2.6. Limpeza de dados
 - 2.6.1. Fases de limpeza de dados
 - 2.6.2. Qualidade dos dados
 - 2.6.3. Manipulação de dados (com R)
- 2.7. Análise de dados, interpretação e avaliação dos resultados
 - 2.7.1. Medidas estatísticas
 - 2.7.2. Indicadores de relação
 - 2.7.3. Extração de dados
- 2.8. Armazém de dados (*Datawarehouse*)
 - 2.8.1. Elementos incluídos
 - 2.8.2. Design
 - 2.8.3. Aspetos a considerar
- 2.9. Disponibilidade dos dados
 - 2.9.1. Acesso
 - 2.9.2. Utilidade
 - 2.9.3. Segurança
- 2.10. Aspetos regulamentares
 - 2.10.1. Lei da Proteção de Dados
 - 2.10.2. Boas práticas
 - 2.10.3. Outros aspetos regulamentares

Módulo 3. O dado na Inteligência Artificial

- 3.1. Ciência de dados
 - 3.1.1. A ciência de dados
 - 3.1.2. Ferramentas avançadas para o cientista de dados
- 3.2. Dados, informação e conhecimento
 - 3.2.1. Dados, informação e conhecimento
 - 3.2.2. Tipos de dados
 - 3.2.3. Fontes de dados
- 3.3. Dos dados à informação
 - 3.3.1. Análise de Dados
 - 3.3.2. Tipos de análise
 - 3.3.3. Extração de informação de um *Dataset*
- 3.4. Extração de informação através da visualização
 - 3.4.1. A visualização como ferramenta de análise
 - 3.4.2. Métodos de visualização
 - 3.4.3. Visualização de um conjunto de dados
- 3.5. Qualidade dos dados
 - 3.5.1. Dados de qualidade
 - 3.5.2. Limpeza de dados
 - 3.5.3. Pré-processamento básico de dados
- 3.6. *Dataset*
 - 3.6.1. Enriquecimento do *Dataset*
 - 3.6.2. A maldição da dimensionalidade
 - 3.6.3. Modificação do nosso conjunto de dados
- 3.7. Desequilíbrio
 - 3.7.1. Desequilíbrio de classes
 - 3.7.2. Técnicas de mitigação do desequilíbrio
 - 3.7.3. Equilíbrio de um *Dataset*
- 3.8. Modelos não supervisionados
 - 3.8.1. Modelo não supervisionado
 - 3.8.2. Métodos
 - 3.8.3. Classificação com modelos não supervisionados

- 3.9. Modelos supervisionados
 - 3.9.1. Modelo supervisionado
 - 3.9.2. Métodos
 - 3.9.3. Classificação com modelos supervisionados
- 3.10. Ferramentas e boas práticas
 - 3.10.1. Boas práticas para um cientista de dados
 - 3.10.2. O melhor modelo
 - 3.10.3. Ferramentas úteis

Módulo 4. Exploração de dados. Seleção, pré-processamento e transformação

- 4.1. A inferência estatística
 - 4.1.1. Estatística descritiva vs. Inferência estatística
 - 4.1.2. Procedimentos paramétricos
 - 4.1.3. Procedimentos não paramétricos
- 4.2. Análise exploratória
 - 4.2.1. Análise descritiva
 - 4.2.2. Visualização
 - 4.2.3. Preparação de dados
- 4.3. Preparação de dados
 - 4.3.1. Integração e limpeza de dados
 - 4.3.2. Normalização de dados
 - 4.3.3. Transformando atributos
- 4.4. Os valores perdidos
 - 4.4.1. Tratamento de valores perdidos
 - 4.4.2. Métodos de imputação de máxima verosimilhança
 - 4.4.3. Imputação de valores perdidos utilizando a aprendizagem automática
- 4.5. O ruído dos dados
 - 4.5.1. Classes de ruído e atributos
 - 4.5.2. Filtragem de ruído
 - 4.5.3. O efeito do ruído
- 4.6. A maldição da dimensionalidade
 - 4.6.1. *Oversampling*
 - 4.6.2. *Undersampling*
 - 4.6.3. Redução de dados multidimensionais

- 4.7. De atributos contínuos a discretos
 - 4.7.1. Dados contínuos versus dados discretos
 - 4.7.2. Processo de discretização
- 4.8. Os dados
 - 4.8.1. Seleção de dados
 - 4.8.2. Perspetivas e critérios de seleção
 - 4.8.3. Métodos de seleção
- 4.9. Seleção de instâncias
 - 4.9.1. Métodos para a seleção de instâncias
 - 4.9.2. Seleção de protótipos
 - 4.9.3. Métodos avançados para a seleção de instâncias
- 4.10. Pré-processamento de dados em ambientes Big Data

Módulo 5. Algoritmo e complexidade na Inteligência Artificial

- 5.1. Introdução às estratégias de desenho do algoritmos
 - 5.1.1. Recursividade
 - 5.1.2. Divide e conquista
 - 5.1.3. Outras estratégias
- 5.2. Eficiência e análise dos algoritmos
 - 5.2.1. Medidas de eficiência
 - 5.2.2. Medir o tamanho da entrada
 - 5.2.3. Medir o tempo de execução
 - 5.2.4. Caso pior, melhor e médio
 - 5.2.5. Notação assintótica
 - 5.2.6. Critérios de Análise matemática de algoritmos não recursivos
 - 5.2.7. Análise matemática de algoritmos recursivos
 - 5.2.8. Análise empírica de algoritmos
- 5.3. Algoritmos de ordenação
 - 5.3.1. Conceito de ordenação
 - 5.3.2. Ordenação da bolha
 - 5.3.3. Ordenação por seleção
 - 5.3.4. Ordenação por inserção
 - 5.3.5. Ordenação por mistura (*Merge_Sort*)
 - 5.3.6. Ordenação rápida (*Quicksort*)

- 5.4. Algoritmos com árvores
 - 5.4.1. Conceito de árvore
 - 5.4.2. Árvores binários
 - 5.4.3. Caminhos de árvore
 - 5.4.4. Representar expressões
 - 5.4.5. Árvores binários ordenadas
 - 5.4.6. Árvores binárias equilibradas
- 5.5. Algoritmos com *Heaps*
 - 5.5.1. Os *Heaps*
 - 5.5.2. O algoritmo *Heapsort*
 - 5.5.3. As filas de prioridade
- 5.6. Algoritmos com Grafos
 - 5.6.1. Representação
 - 5.6.2. Caminho de largura
 - 5.6.3. Caminho de profundidade
 - 5.6.4. Ordenação topológica
- 5.7. Algoritmos *Greedy*
 - 5.7.1. A estratégia *Greedy*
 - 5.7.2. Elementos da estratégia *Greedy*
 - 5.7.3. Câmbio de moedas
 - 5.7.4. Problema do viajante
 - 5.7.5. Problema da mochila
- 5.8. Pesquisa de caminhos mínimos
 - 5.8.1. O problema do caminho mínimo
 - 5.8.2. Arcos negativos e ciclos
 - 5.8.3. Algoritmo de Dijkstra
- 5.9. Algoritmos *Greedy* sobre Grafos
 - 5.9.1. A árvore de extensão mínima
 - 5.9.2. O algoritmo de Prim
 - 5.9.3. O algoritmo Kruskal
 - 5.9.4. Análise de complexidade
- 5.10. *Backtracking*
 - 5.10.1. O *Backtracking*
 - 5.10.2. Técnicas alternativas

Módulo 6. Sistemas inteligentes

- 6.1. Teoria dos agentes
 - 6.1.1. História do conceito
 - 6.1.2. Definição de agente
 - 6.1.3. Agentes na Inteligência Artificial
 - 6.1.4. Agentes em Engenharia de Software
- 6.2. Arquiteturas de agentes
 - 6.2.1. O processo de argumentação de um agente
 - 6.2.2. Agentes reativos
 - 6.2.3. Agentes dedutivos
 - 6.2.4. Agentes híbridos
 - 6.2.5. Comparativo
- 6.3. Informação e conhecimento
 - 6.3.1. Distinção entre dados, informação e conhecimento
 - 6.3.2. Avaliação qualidade dos dados
 - 6.3.3. Métodos de recolha de dados
 - 6.3.4. Métodos de aquisição de dados
 - 6.3.5. Métodos de aquisição de conhecimento
- 6.4. Representação do conhecimento
 - 6.4.1. A importância da representação do conhecimento
 - 6.4.2. Definição da representação do conhecimento através das suas funções
 - 6.4.3. Características de uma representação do conhecimento
- 6.5. Ontologias
 - 6.5.1. Introdução aos metadados
 - 6.5.2. Conceito filosófico de ontologia
 - 6.5.3. Conceito informático de ontologia
 - 6.5.4. Ontologias de domínio e ontologias de nível superior
 - 6.5.5. Como construir uma ontologia?
- 6.6. Linguagens para ontologias e Software para a criação de ontologias
 - 6.6.1. Triples RDF, *Turtle* e N
 - 6.6.2. RDF *Schema*
 - 6.6.3. OWL

- 6.6.4. SPARQL
- 6.6.5. Introdução às diferentes ferramentas de criação de ontologias
- 6.6.6. Instalação e utilização do *Protégé*
- 6.7. A web semântica
 - 6.7.1. O estado atual e futuro da web semântica
 - 6.7.2. Aplicações da web semântica
- 6.8. Outros modelos representação do conhecimento
 - 6.8.1. Vocabulários
 - 6.8.2. Visão global
 - 6.8.3. Taxonomias
 - 6.8.4. Tesaurus
 - 6.8.5. Folksonomias
 - 6.8.6. Comparativo
 - 6.8.7. Mapas mentais
- 6.9. Avaliação e integração das representações do conhecimento
 - 6.9.1. Lógica de ordem zero
 - 6.9.2. Lógica de primeira ordem
 - 6.9.3. Lógica descritiva
 - 6.9.4. Relação entre diferentes tipos de lógica
 - 6.9.5. *Prolog*: programação baseada na lógica de primeira ordem
- 6.10. Raciocinadores semânticos, sistemas baseados no conhecimento e Sistemas Periciais
 - 6.10.1. Conceito de raciocinador
 - 6.10.2. Aplicações de um raciocinador
 - 6.10.3. Sistemas baseados no conhecimento
 - 6.10.4. MYCIN, história dos Sistemas Periciais
 - 6.10.5. Elementos e Arquitetura dos Sistemas Periciais
 - 6.10.6. Criação de Sistemas Periciais

Módulo 7. Aprendizagem automática e mineração de dados

- 7.1. Introdução aos processos de descoberta de conhecimentos e aos conceitos básicos da aprendizagem automática
 - 7.1.1. Conceitos-chave dos processos de descoberta do conhecimento
 - 7.1.2. Perspetiva histórica dos processos de descoberta do conhecimento
 - 7.1.3. Etapas dos processos de descoberta do conhecimento

- 7.1.4. Técnicas utilizadas nos processos de descoberta do conhecimento
- 7.1.5. Características dos bons modelos de aprendizagem automática
- 7.1.6. Tipos de informação sobre aprendizagem automática
- 7.1.7. Conceitos básicos de aprendizagem
- 7.1.8. Conceitos básicos de aprendizagem não supervisionado
- 7.2. Exploração e pré-processamento de dados
 - 7.2.1. Tratamento de dados
 - 7.2.2. Tratamento de dados no fluxo de análise de dados
 - 7.2.3. Tipos de dados
 - 7.2.4. Transformação de dados
 - 7.2.5. Visualização e exploração de variáveis contínuas
 - 7.2.6. Visualização e exploração de variáveis categóricas
 - 7.2.7. Medidas de correlação
 - 7.2.8. Representações gráficas mais comuns
 - 7.2.9. Introdução à análise multivariada e à redução da dimensionalidade
- 7.3. Árvore de decisão
 - 7.3.1. Algoritmo ID
 - 7.3.2. Algoritmo C
 - 7.3.3. Excesso de treino e poda
 - 7.3.4. Análise dos resultados
- 7.4. Avaliação dos classificadores
 - 7.4.1. Matrizes de confusão
 - 7.4.2. Matrizes de avaliação numérica
 - 7.4.3. Estatística Kappa
 - 7.4.4. A curva ROC
- 7.5. Regras de classificação
 - 7.5.1. Medidas de avaliação das regras
 - 7.5.2. Introdução à representação gráfica
 - 7.5.3. Algoritmo de sobreposição sequencial
- 7.6. Redes neurais
 - 7.6.1. Conceitos básicos
 - 7.6.2. Redes neuronais simples
 - 7.6.3. Algoritmo de *Backpropagation*
 - 7.6.4. Introdução às redes neuronais recorrentes

- 7.7. Métodos bayesianos
 - 7.7.1. Conceitos básicos de probabilidade
 - 7.7.2. Teorema de Bayes
 - 7.7.3. Naive Bayes
 - 7.7.4. Introdução às redes bayesianas
- 7.8. Modelos de regressão e modelos de resposta contínua
 - 7.8.1. Regressão linear simples
 - 7.8.2. Regressão linear múltipla
 - 7.8.3. Regressão logística
 - 7.8.4. Árvores de regressão
 - 7.8.5. Introdução às máquinas de suporte vetorial (SVM)
 - 7.8.6. Medidas de adequação
- 7.9. *Clustering*
 - 7.9.1. Conceitos básicos
 - 7.9.2. *Clustering* hierárquico
 - 7.9.3. Métodos probabilísticos
 - 7.9.4. Algoritmo EM
 - 7.9.5. Método *B-Cubed*
 - 7.9.6. Métodos implícitos
- 7.10. Mineração de texto e processamento linguagem natural(PLN)
 - 7.10.1. Conceitos básicos
 - 7.10.2. Criação do corpus
 - 7.10.3. Análise descritiva
 - 7.10.4. Introdução à análise de sentimentos

Módulo 8. As redes neuronais, a base da *Deep Learning*

- 8.1. Aprendizagem Profunda
 - 8.1.1. Tipos de aprendizagem profunda
 - 8.1.2. Aplicações da aprendizagem profunda
 - 8.1.3. Vantagens e desvantagens da aprendizagem profunda
- 8.2. Operações
 - 8.2.1. Adição
 - 8.2.2. Produto
 - 8.2.3. Transferência
- 8.3. Camadas
 - 8.3.1. Camada de entrada
 - 8.3.2. Camada oculta
 - 8.3.3. Camada de saída
- 8.4. Ligação de Camadas e Operações
 - 8.4.1. Design de arquiteturas
 - 8.4.2. Conexão entre camadas
 - 8.4.3. Propagação para a frente
- 8.5. Construção da primeira rede neuronal
 - 8.5.1. Design da rede
 - 8.5.2. Estabelecer os pesos
 - 8.5.3. Treino da rede
- 8.6. Treinador e Otimizador
 - 8.6.1. Seleção do otimizador
 - 8.6.2. Estabelecimento de uma função de perda
 - 8.6.3. Estabelecimento de uma métrica
- 8.7. Aplicação dos Princípios das Redes Neuronais
 - 8.7.1. Funções de ativação
 - 8.7.2. Propagação para trás
 - 8.7.3. Ajuste dos parâmetros
- 8.8. Dos neurónios biológicos aos neurónios artificiais
 - 8.8.1. Funcionamento de um neurónio biológico
 - 8.8.2. Transferência de conhecimentos para os neurónios artificiais
 - 8.8.3. Estabelecer de relações entre os dois
- 8.9. Implementação do MLP (Perceptron Multicamadas) com o Keras
 - 8.9.1. Definição da estrutura da rede
 - 8.9.2. Compilação do modelo
 - 8.9.3. Treino do modelo
- 8.10. Hiperparâmetros de *Fine tuning* de Redes Neuronais
 - 8.10.1. Seleção da função de ativação
 - 8.10.2. Estabelecer a *Learning rate*
 - 8.10.3. Ajuste dos pesos

Módulo 9. Treino de redes neurais profundas

- 9.1. Problemas de Gradientes
 - 9.1.1. Técnicas de otimização de gradiente
 - 9.1.2. Gradientes Estocásticos
 - 9.1.3. Técnicas de inicialização de pesos
- 9.2. Reutilização de camadas pré-treinadas
 - 9.2.1. Treino de transferência de aprendizagem
 - 9.2.2. Extração de características
 - 9.2.3. Aprendizagem profunda
- 9.3. Otimizadores
 - 9.3.1. Otimizadores estocásticos de gradiente descendente
 - 9.3.2. Otimizadores Adam e RMSprop
 - 9.3.3. Otimizadores de momento
- 9.4. Programação da taxa de aprendizagem
 - 9.4.1. Controlo de taxa sobre aprendizagem automática
 - 9.4.2. Ciclos de aprendizagem
 - 9.4.3. Termos de suavização
- 9.5. Sobreajuste
 - 9.5.1. Validação cruzada
 - 9.5.2. Regularização
 - 9.5.3. Métricas de avaliação
- 9.6. Orientações práticas
 - 9.6.1. Design do modelo
 - 9.6.2. Seleção de métricas e parâmetros de avaliação
 - 9.6.3. Teste de hipóteses
- 9.7. *Transfer Learning*
 - 9.7.1. Treino de transferência de aprendizagem
 - 9.7.2. Extração de características
 - 9.7.3. Aprendizagem profunda
- 9.8. *Data Augmentation*
 - 9.8.1. Transformações de imagem
 - 9.8.2. Geração de dados sintéticos
 - 9.8.3. Transformação de texto

- 9.9. Aplicação Prática de *Transfer Learning*
 - 9.9.1. Treino de transferência de aprendizagem
 - 9.9.2. Extração de características
 - 9.9.3. Aprendizagem profunda
- 9.10. Regularização
 - 9.10.1. L e L
 - 9.10.2. Regularização por entropia máxima
 - 9.10.3. *Dropout*

Módulo 10. Personalização de Modelos e treino com *TensorFlow*

- 10.1. *TensorFlow*
 - 10.1.1. Uso da biblioteca *TensorFlow*
 - 10.1.2. Treino de modelos com o *TensorFlow*
 - 10.1.3. Operações de gráfico no *TensorFlow*
- 10.2. *TensorFlow* e NumPy
 - 10.2.1. Ambiente computacional NumPy para *TensorFlow*
 - 10.2.2. Utilização das arrays NumPy com o *TensorFlow*
 - 10.2.3. Operações NumPy para o *TensorFlow* gráficos do *TensorFlow*
- 10.3. Personalização de modelos e algoritmos de treino
 - 10.3.1. Construir modelos personalizados com o *TensorFlow*
 - 10.3.2. Gestão dos parâmetros de treino
 - 10.3.3. Utilização de técnicas de otimização para o treino
- 10.4. Funções e gráficos do *TensorFlow*
 - 10.4.1. Funções com o *TensorFlow*
 - 10.4.2. Utilização de gráficos para treino de modelos
 - 10.4.3. Otimização de gráficos com operações do *TensorFlow*
- 10.5. Carga de conjuntos de dados com o *TensorFlow*
 - 10.5.1. Carga de conjuntos de dados com o *TensorFlow*
 - 10.5.2. Pré-processamento de dados com o *TensorFlow*
 - 10.5.3. Utilizar de ferramentas do *TensorFlow* para a manipulação de dados
- 10.6. A API *tfdata*
 - 10.6.1. Utilização da API *tfdata* para o processamento de dados
 - 10.6.2. Construção de fluxo de dados com *tfdata*
 - 10.6.3. Utilização da API *tfdata* para o treino de modelos

- 10.7. O formato *TFRecord*
 - 10.7.1. Utilização da API *TFRecord* para a serialização de dados
 - 10.7.2. Carregar arquivos *TFRecord* com *TensorFlow*
 - 10.7.3. Utilização de arquivos *TFRecord* para o treino de modelos
 - 10.8. Camadas de pré-processamento do Keras
 - 10.8.1. Utilização da API de pré-processamento do Keras
 - 10.8.2. Construção de *pipelined* de pré-processamento com o Keras
 - 10.8.3. Utilização da API de pré-processamento do Keras para o treino de modelos
 - 10.9. O projeto *TensorFlow Datasets*
 - 10.9.1. Utilização de *TensorFlow Datasets* para o carregamento de dados
 - 10.9.2. Pré-processamento de dados com o *TensorFlow Datasets*
 - 10.9.3. Utilização *TensorFlow Datasets* para o treino de modelos
 - 10.10. Construção de uma Aplicação de Deep *Learning* com *TensorFlow*
 - 10.10.1. Aplicação Prática
 - 10.10.2. Construção de uma aplicação de Deep *Learning* com *TensorFlow*
 - 10.10.3. Treino de um modelo com o *TensorFlow*
 - 10.10.4. Utilizar a aplicação para previsão de resultados
- Módulo 11. Deep Computer Vision com Redes Neurais Convolucionais**
- 11.1. A Arquitetura *Visual Cortex*
 - 11.1.1. Funções do córtex visual
 - 11.1.2. Teoria da visão computacional
 - 11.1.3. Modelos de processamento de imagens
 - 11.2. Camadas convolucionais
 - 11.2.1. Reutilização de pesos na convolução
 - 11.2.2. Convolução D
 - 11.2.3. Funções de ativação
 - 11.3. Camadas de agrupamento e implementação de camadas de agrupamento
 - 11.3.1. *Pooling* e *Striding*
 - 11.3.2. *Flattening*
 - 11.3.3. Tipos de *Pooling*
 - 11.4. Arquitetura CNN
 - 11.4.1. Arquitetura VGG
 - 11.4.2. Arquitetura *AlexNet*
 - 11.4.3. Arquitetura *ResNet*
 - 11.5. Implementação de uma CNN ResNet- usando Keras
 - 11.5.1. Inicialização de pesos
 - 11.5.2. Definição da camada de entrada
 - 11.5.3. Definição da saída
 - 11.6. Utilização de modelos pré-treinados do Keras
 - 11.6.1. Características dos modelos pré-treinados
 - 11.6.2. Usos dos modelos pré-treinados
 - 11.6.3. Vantagens dos modelos pré-treinados
 - 11.7. Modelos pré-treinados para a aprendizagem por transferência
 - 11.7.1. A Aprendizagem por transferência
 - 11.7.2. Processo de aprendizagem por transferência
 - 11.7.3. Vantagens do aprendizagem por transferência
 - 11.8. Classificação e Localização em *Deep Computer Vision*
 - 11.8.1. Classificação de imagens
 - 11.8.2. Localização de objetos em imagens
 - 11.8.3. Detecção de objetos
 - 11.9. Detecção e seguimento de objetos
 - 11.9.1. Métodos de detecção de objetos
 - 11.9.2. Algoritmos de seguimento de objetos
 - 11.9.3. Técnicas de seguimento e localização
 - 11.10. Segmentação semântica
 - 11.10.1. Aprendizagem profunda para a segmentação semântica
 - 11.10.2. Detecção de bordas
 - 11.10.3. Métodos de segmentação baseado sem regras

Módulo 12. Processamento de linguagem natural (PLN) com Redes Neurais Recorrentes (RNN) e Atenção

- 12.1. Geração de texto utilizando RNN
 - 12.1.1. Treino de uma RNN para geração de texto
 - 12.1.2. Geração de linguagem natural com RNN
 - 12.1.3. Aplicações de geração de texto com RNN
- 12.2. Criação de conjuntos de dados de treino
 - 12.2.1. Preparação dos dados para o treino de uma RNN
 - 12.2.2. Armazenamento do conjunto de dados de treino
 - 12.2.3. Limpeza e transformação dos dados
 - 12.2.4. Análise de Sentimento
- 12.3. Classificação da opiniões com RNN
 - 12.3.1. Detecção de temas nos comentários
 - 12.3.2. Análise de sentimento com algoritmos de aprendizagem profunda
- 12.4. Rede codificadora-descodificadora para a tradução automática neural
 - 12.4.1. Treino de uma RNN para a tradução automática
 - 12.4.2. Utilização de uma rede *encoder-decoder* para a tradução automática
 - 12.4.3. Melhoria da precisão da tradução automática com RNNs
- 12.5. Mecanismos de atenção
 - 12.5.1. Implementação de mecanismos de atenção em RNN
 - 12.5.2. Utilização de mecanismos de atenção para melhorar a precisão dos modelos
 - 12.5.3. Vantagens dos mecanismos de atenção nas redes neuronais
- 12.6. Modelos *Transformers*
 - 12.6.1. Uso de modelos *Transformers* no processamento de linguagem natural
 - 12.6.2. Aplicação de modelos *Transformers* na visão
 - 12.6.3. Vantagens dos modelos *Transformers*
- 12.7. *Transformers* para a visão
 - 12.7.1. Utilização de modelos *Transformers* para a visão
 - 12.7.2. Pré-processamento de dados de imagem
 - 12.7.3. Treino de modelos *Transformers* para visão
- 12.8. Biblioteca de *Transformers* de *Hugging Face*
 - 12.8.1. Utilização da biblioteca *Transformers* de *Hugging Face*
 - 12.8.2. Aplicação da biblioteca de *Transformers* de *Hugging Face*
 - 12.8.3. Vantagens da biblioteca *Transformers* de *Hugging Face*

- 12.9. Outras Bibliotecas de *Transformers*. Comparativo
 - 12.9.1. Comparação entre as diferentes bibliotecas de *Transformers*
 - 12.9.2. Uso das outras bibliotecas de *Transformers*
 - 12.9.3. Vantagens das outras bibliotecas de *Transformers*
- 12.10. Desenvolvimento de uma aplicação de PLN com RNN e Atenção. Aplicação Prática
 - 12.10.1. Desenvolvimento de uma aplicação de processamento de linguagem natural com RNN e atenção
 - 12.10.2. Utilização de RNN, mecanismos de atenção e modelos *Transformers* na aplicação
 - 12.10.3. Avaliação da aplicação prática

Módulo 13. Autoencoders, GANs, e Modelos de Difusão

- 13.1. Representação de dados eficientes
 - 13.1.1. Redução da dimensionalidade
 - 13.1.2. Aprendizagem profunda
 - 13.1.3. Representações compactas
- 13.2. Realização da PCA com um codificador automático linear incompleto
 - 13.2.1. Processo de treino
 - 13.2.2. Implementação em Python
 - 13.2.3. Utilização de dados de teste
- 13.3. Codificadores automáticos empilhados
 - 13.3.1. Redes neuronais profundas
 - 13.3.2. Construção de arquiteturas de codificação
 - 13.3.3. Utilização da regularização
- 13.4. Autoencodificadores convolucionais
 - 13.4.1. Design do modelo convolucionais
 - 13.4.2. Treino do modelo convolucionais
 - 13.4.3. Avaliação dos resultados
- 13.5. Redução do ruído dos codificadores automáticos
 - 13.5.1. Aplicação de filtros
 - 13.5.2. Design de modelos de codificação
 - 13.5.3. Utilização de técnicas de regularização
- 13.6. Codificadores automáticos dispersos
 - 13.6.1. Aumento da eficiência da codificação
 - 13.6.2. Minimizar o número de parâmetros
 - 13.6.3. Utilização de técnicas de regularização

- 13.7. Codificadores automáticos variacionais
 - 13.7.1. Utilização da otimização variacional
 - 13.7.2. Aprendizagem profunda não supervisionada
 - 13.7.3. Representações latentes profundas
- 13.8. Geração de imagens MNIST de moda
 - 13.8.1. Reconhecimento de padrões
 - 13.8.2. Geração de imagens
 - 13.8.3. Treino de redes neurais profundas
- 13.9. Redes generativas antagônicas e modelos de difusão
 - 13.9.1. Geração de conteúdos a partir de imagens
 - 13.9.2. Modelação de distribuições de dados
 - 13.9.3. Utilização de redes contraditórias
- 13.10. Implementação dos Modelos
 - 13.10.1. Aplicação Prática
 - 13.10.2. Implementação dos modelos
 - 13.10.3. Utilização de dados reais
 - 13.10.4. Avaliação dos resultados

Módulo 14. Computação bioinspirada

- 14.1. Introdução à computação bioinspirada
 - 14.1.1. Introdução à computação bioinspirada
- 14.2. Algoritmos de inspiração social
 - 14.2.1. Computação bioinspirada baseada em colônias de formigas
 - 14.2.2. Variantes dos algoritmos de colônias de formigas
 - 14.2.3. Computação baseada em nuvens de partículas
- 14.3. Algoritmos genéticos
 - 14.3.1. Estrutura geral
 - 14.3.2. Implementações dos principais operadores
- 14.4. Estratégias de exploração do espaço para algoritmos genéticos
 - 14.4.1. Algoritmo CHC
 - 14.4.2. Problemas multimodais

- 14.5. Modelos de computação evolutiva
 - 14.5.1. Estratégias evolutivas
 - 14.5.2. Programação evolutiva
 - 14.5.3. Algoritmos baseados em evolução diferencial
- 14.6. Modelos de computação evolutiva (II)
 - 14.6.1. Modelos de evolução baseados na estimativa das distribuições (EDA)
 - 14.6.2. Programação genética
- 14.7. Programação evolutiva aplicada a problemas de aprendizagem
 - 14.7.1. A aprendizagem baseada em regras
 - 14.7.2. Métodos evolutivos em problemas de seleção de exemplos
- 14.8. Problemas multiobjetivo
 - 14.8.1. Conceito de dominância
 - 14.8.2. Aplicação de algoritmos evolutivos a problemas multiobjetivos
- 14.9. Redes neurais (I)
 - 14.9.1. Introdução às redes neurais
 - 14.9.2. Exemplo prático com redes neurais
- 14.10. Redes neurais (II)
 - 14.10.1. Casos de utilização de redes neurais na investigação médica
 - 14.10.2. Casos de utilização de redes neurais na economia
 - 14.10.3. Casos de utilização de redes neurais na visão artificial

Módulo 15. Inteligência Artificial Estratégias e aplicações

- 15.1. Serviços financeiros
 - 15.1.1. As implicações da Inteligência Artificial (IA) nos serviços financeiros Oportunidades e desafios
 - 15.1.2. Casos de utilização
 - 15.1.3. Potenciais riscos relacionados com a utilização de IA
 - 15.1.4. Potenciais desenvolvimentos/utilizações futuras da IA
- 15.2. Implicações da inteligência artificial no serviço de saúde
 - 15.2.1. Implicações da IA no setor da saúde Oportunidades e desafios
 - 15.2.2. Casos de utilização
- 15.3. Riscos relacionados com a utilização de IA no serviço de saúde
 - 15.3.1. Potenciais riscos relacionados com a utilização de IA
 - 15.3.2. Potenciais desenvolvimentos/utilizações futuras da IA

- 15.4. *Retail*
 - 15.4.1. Implicações da IA no *Retail* Oportunidades e desafios
 - 15.4.2. Casos de utilização
 - 15.4.3. Potenciais riscos relacionados com a utilização de IA
 - 15.4.4. Potenciais desenvolvimentos/utilizações futuras da IA
- 15.5. Indústrias
 - 15.5.1. Implicações da IA na Indústria. Oportunidades e desafios
 - 15.5.2. Casos de utilização
- 15.6. Potenciais riscos relacionados com a utilização de IA na Indústria
 - 15.6.1. Casos de utilização
 - 15.6.2. Potenciais riscos relacionados com a utilização de IA
 - 15.6.3. Potenciais desenvolvimentos/utilizações futuras da IA
- 15.7. Administração pública
 - 15.7.1. Implicações da IA na Administração pública. Oportunidades e desafios
 - 15.7.2. Casos de utilização
 - 15.7.3. Potenciais riscos relacionados com a utilização de IA
 - 15.7.4. Potenciais desenvolvimentos/utilizações futuras da IA
- 15.8. Educação
 - 15.8.1. Implicações da IA na educação. Oportunidades e desafios
 - 15.8.2. Casos de utilização
 - 15.8.3. Potenciais riscos relacionados com a utilização de IA
 - 15.8.4. Potenciais desenvolvimentos/utilizações futuras da IA
- 15.9. Silvicultura e agricultura
 - 15.9.1. Implicações da IA na Indústria 4.0 Oportunidades e desafios
 - 15.9.2. Casos de utilização
 - 15.9.3. Potenciais riscos relacionados com a utilização de IA
 - 15.9.4. Potenciais desenvolvimentos/utilizações futuras da IA
- 15.10 Recursos Humanos
 - 15.10.1. Implicações da IA nos Recursos Humanos Oportunidades e desafios
 - 15.10.2. Casos de utilização
 - 15.10.3. Potenciais riscos relacionados com a utilização de IA
 - 15.10.4. Potenciais desenvolvimentos/utilizações futuras da IA

Módulo 16. Melhorar a produtividade no Desenvolvimento de software com IA

- 16.1. Preparar um ambiente de desenvolvimento adequado
 - 16.1.1. Seleção de ferramentas essenciais para o desenvolvimento com IA
 - 16.1.2. Configuração das ferramentas selecionadas
 - 16.1.3. Implementação de pipelines CI/CD adaptados a projetos com IA
 - 16.1.4. Gestão eficiente de dependências e versões em ambientes de desenvolvimento
- 16.2. Extensões imprescindíveis de IA para o Visual Studio Code
 - 16.2.1. Exploração e seleção de extensões de IA para o Visual Studio Code
 - 16.2.2. Integração de ferramentas de análise estática e dinâmica na IDE
 - 16.2.3. Automatização de tarefas repetitivas com extensões específicas
 - 16.2.4. Personalização do ambiente de desenvolvimento para melhorar a eficiência
- 16.3. Design No-code de Interfaces de Utilizador com Flutterflow
 - 16.3.1. Princípios de design *No-code* e a sua aplicação em interfaces de utilizador
 - 16.3.2. Incorporação de elementos de IA no design de interfaces visuais
 - 16.3.3. Ferramentas e plataformas para a criação *No-code* de interfaces inteligentes
 - 16.3.4. Avaliação e melhoria contínua de interfaces *No-code* com IA
- 16.4. Otimização do código utilizando o ChatGPT
 - 16.4.1. Identificar código duplicado
 - 16.4.2. Refatorar
 - 16.4.3. Criar códigos legíveis
 - 16.4.4. Compreender o que o código faz
 - 16.4.5. Melhorar dos nomes de variáveis e funções
 - 16.4.6. Criação de documentação automática
- 16.5. Gestão de repositórios com IA utilizando o ChatGPT
 - 16.5.1. Automatização dos processos de controlo de versões com técnicas de IA
 - 16.5.2. Detecção e resolução automática de conflitos em ambientes de colaboração
 - 16.5.3. Análise preditiva de alterações e tendências em repositórios de código
 - 16.5.4. Melhorias na organização e categorização dos repositórios utilizando a IA
- 16.6. Integração da IA na gestão de bases de dados com o AskYourDatabase
 - 16.6.1. Otimização de consultas e desempenho utilizando técnicas de IA
 - 16.6.2. Análise preditiva dos padrões de acesso à base de dados
 - 16.6.3. Implementação de sistemas de recomendação para otimizar a estrutura da base de dados
 - 16.6.4. Monitorização e deteção proativa de potenciais problemas na base de dados

- 16.7. Procura de falhas e criação de testes unitários com IA usando o ChatGPT
 - 16.7.1. Geração automática de casos de teste utilizando técnicas de IA
 - 16.7.2. Detecção precoce de vulnerabilidades e erros utilizando a análise estática com IA
 - 16.7.3. Melhora da cobertura dos testes através da identificação de áreas críticas pela IA
 - 16.8. *Pair Programming* com GitHub Copilot
 - 16.8.1. Integração e utilização efectiva do GitHub Copilot em sessões de *Pair Programming*
 - 16.8.2. Integração Melhoras na comunicação e a colaboração dos desenvolvedores com o GitHub Copilot
 - 16.8.3. Integração Estratégias para tirar o máximo partido das sugestões de código geradas pelo GitHub Copilot
 - 16.8.4. Integração Estudos de Caso e boas práticas em *Pair Programming* assistido por IA
 - 16.9. Tradução automática entre linguagens de programação utilizando o ChatGPT
 - 16.9.1. Ferramentas e serviços de tradução automática específicos para linguagens de programação
 - 16.9.2. Adaptação de algoritmos de tradução automática a contextos de desenvolvimento
 - 16.9.3. Melhora a interoperabilidade entre diferentes linguagens através da tradução automática
 - 16.9.4. Avaliação e atenuação dos potenciais desafios e limitações na tradução automática
 - 16.10. Ferramentas de IA recomendadas para melhorar a produtividade
 - 16.10.1. Análise comparativa das ferramentas de IA para o desenvolvimento de software
 - 16.10.2. Integração das ferramentas de IA em fluxos de trabalho
 - 16.10.3. Automatização de tarefas de rotina com ferramentas de IA
 - 16.10.4. Avaliação e seleção de ferramentas com base no contexto e nos requisitos do projeto
- Módulo 17. Arquitetura de software com IA**
- 17.1. Otimização e gestão do desempenho em ferramentas com IA com a ajuda do ChatGPT
 - 17.1.1. Análise e perfilagem de desempenho em ferramentas com IA
 - 17.1.2. Estratégias de otimização de algoritmos e modelos de IA
 - 17.1.3. Implementação de técnicas de *caching* e paralelização para melhorar o desempenho
 - 17.1.4. Ferramentas e metodologias para a monitorização contínua do desempenho em tempo real
 - 17.2. Escalabilidade em aplicações de IA usando o ChatGPT
 - 17.2.1. Design de arquitecturas escaláveis para aplicações de IA
 - 17.2.2. Implementação de técnicas de particionamento e distribuição de carga
 - 17.2.3. Gestão de fluxos de trabalho e carga de trabalho em sistemas escaláveis
 - 17.2.4. Estratégias para a expansão horizontal e vertical em ambientes com demanda variável
 - 17.3. Manutenibilidade de aplicações com IA usando o ChatGPT
 - 17.3.1. Princípios de design para facilitar a manutenibilidade em projetos de IA
 - 17.3.2. Estratégias de documentação específicas para modelos e algoritmos de IA
 - 17.3.3. Implementação de testes unitários e de integração para facilitar a manutenção
 - 17.3.4. Métodos para a refatoração e melhoria contínua em sistemas com componentes de IA
 - 17.4. Design de sistemas de grande escala
 - 17.4.1. Princípios arquitetónicos para o design de sistemas de grande escala
 - 17.4.2. Desagregação de sistemas complexos em microserviços
 - 17.4.3. Implementação de padrões de design específicos para sistemas distribuídos
 - 17.4.4. Estratégias para a gestão da complexidade em arquitecturas de grande escala com componentes de IA
 - 17.5. Armazenamento de dados de grande escala para ferramentas de IA
 - 17.5.1. Seleção de tecnologias de armazenamento de dados escaláveis
 - 17.5.2. Design de esquemas de bases de dados para a gestão eficiente de grandes volumes de dados
 - 17.5.3. Estratégias de particionamento e replicação em ambientes de armazenamento de dados massivos
 - 17.5.4. Implementação de sistemas de gestão de dados para garantir a integridade e disponibilidade em projetos com IA
 - 17.6. Estruturas de dados com IA usando o ChatGPT
 - 17.6.1. Adaptação de estruturas de dados clássicas para o seu uso em algoritmos de IA
 - 17.6.2. Design e otimização de estruturas de dados específicas com o ChatGPT
 - 17.6.3. Integração de estruturas de dados eficientes em sistemas com processamento intensivo de dados
 - 17.6.4. Estratégias para a manipulação e armazenamento de dados em tempo real em estruturas de dados com IA

- 17.7. Algoritmos de programação para produtos com IA
 - 17.7.1. Desenvolvimento e implementação de algoritmos específicos para aplicações com IA
 - 17.7.2. Estratégias de seleção de algoritmos conforme o tipo de problema e os requisitos do produto
 - 17.7.3. Adaptação de algoritmos clássicos para a sua integração em sistemas de inteligência artificial
 - 17.7.4. Avaliação e comparação de desempenho entre diferentes algoritmos em contextos de desenvolvimento com IA
- 17.8. Padrões de design para desenvolvimento com IA
 - 17.8.1. Identificação e aplicação de padrões de design comuns em projetos com componentes de IA
 - 17.8.2. Desenvolvimento de padrões específicos para a integração de modelos e algoritmos em sistemas existentes
 - 17.8.3. Estratégias de implementação de padrões para melhorar a reutilização e manutenibilidade em projetos de IA
 - 17.8.4. Casos de estudo e boas práticas na aplicação de padrões de design em arquiteturas com IA
- 17.9. Implementação de clean architecture usando o ChatGPT
 - 17.9.1. Princípios e conceitos fundamentais de *Clean Architecture*
 - 17.9.2. Adaptação de *Clean Architecture* para projetos com componentes de IA
 - 17.9.3. Implementação de camadas e dependências em sistemas com arquitetura limpa
 - 17.9.4. Benefícios e desafios da implementação de *Clean Architecture* no desenvolvimento de software com IA
- 17.10. Desenvolvimento de software seguro em aplicações web com DeepCode
 - 17.10.1. Princípios de segurança no desenvolvimento de software com componentes de IA
 - 17.10.2. Identificação e atenuação de potenciais vulnerabilidades nos modelos e algoritmos de IA
 - 17.10.3. Implementação de práticas de desenvolvimento seguro em aplicações web com funcionalidades de Inteligência Artificial
 - 17.10.4. Estratégias para a proteção de dados sensíveis e a prevenção de ataques em projetos com IA

Módulo 18. Projetos Web com IA

- 18.1. Preparar o Ambiente de Trabalho para o Desenvolvimento Web com IA
 - 18.1.1. Configuração de ambientes de desenvolvimento web para projetos com inteligência artificial
 - 18.1.2. Seleção e preparação de ferramentas essenciais para o desenvolvimento web com IA
 - 18.1.3. Integração de bibliotecas e *frameworks* específicos para projetos web com inteligência artificial
 - 18.1.4. Implementação das melhores práticas na configuração de ambientes de desenvolvimento colaborativos
- 18.2. Criação de *Workspace* para projetos de IA
 - 18.2.1. Design e organização eficaz de *workspaces* para projectos web com componentes de inteligência artificial
 - 18.2.2. A utilização de ferramentas de gestão de projetos e de controlo de versões no *workspace*
 - 18.2.3. Estratégias para a colaboração e comunicação eficientes na equipa de desenvolvimento
 - 18.2.4. Adaptação do *workspace* às necessidades específicas dos projetos web com IA
- 18.3. Padrões de Design em Produtos com GitHub Copilot
 - 18.3.1. Identificação e aplicação de padrões de design comuns em interfaces de utilizador com elementos de inteligência artificial
 - 18.3.2. Desenvolvimento de padrões específicos para melhorar a experiência do utilizador em projetos web com IA
 - 18.3.3. Integração de padrões de design na arquitetura geral de projetos web com Inteligência Artificial
 - 18.3.4. Avaliação e seleção de padrões de design adequados conforme o contexto do projeto
- 18.4. Desenvolvimento Frontend com GitHub Copilot
 - 18.4.1. Integração de modelos de IA na camada de apresentação de projetos web
 - 18.4.2. Desenvolvimento de interfaces de utilizador adaptativas com elementos de inteligência artificial
 - 18.4.3. Implementação de funcionalidades de processamento de linguagem natural (PLN) no Frontend
 - 18.4.4. Estratégias para a otimização do desempenho no desenvolvimento Frontend com IA

- 18.5. Criação de Base de Dados usando GitHub Copilot
 - 18.5.1. Seleção de tecnologias de bases de dados para projetos web com inteligência artificial
 - 18.5.2. Design de esquemas de bases de dados para armazenar e gerir dados relacionados com IA
 - 18.5.3. Implementação de sistemas de armazenamento eficientes para grandes volumes de dados gerados por modelos de IA
 - 18.5.4. Estratégias para a segurança e proteção de dados sensíveis em bases de dados de projetos web com IA
 - 18.6. Desenvolvimento Backend com GitHub Copilot
 - 18.6.1. Integração de serviços e modelos de IA na lógica de negócios do Backend
 - 18.6.2. Desenvolvimento de APIs e endpoints específicos para a comunicação entre o Frontend e os componentes de IA
 - 18.6.3. Implementação de lógica de processamento de dados e tomada de decisões no Backend com Inteligência Artificial
 - 18.6.4. Estratégias para a escalabilidade e desempenho no desenvolvimento Backend de projetos web com IA
 - 18.7. Optimizar o Processo de Desdobramento do Seu Website
 - 18.7.1. Automatização de processos de construção e desdobramento de projetos web com o ChatGPT
 - 18.7.2. Implementação de pipelines de CI/CD adaptados a aplicações web com GitHub Copilot
 - 18.7.3. Estratégias para a gestão eficiente de versões e atualizações em desdobramentos contínuos
 - 18.7.4. Monitorização e análise pós-desdobramento para a melhoria contínua do processo
 - 18.8. IA na Computação em Nuvem
 - 18.8.1. Integração de serviços de inteligência artificial em plataformas de computação em nuvem
 - 18.8.2. Desenvolvimento de soluções escaláveis e distribuídas utilizando serviços de nuvem com capacidades de IA
 - 18.8.3. Estratégias para a gestão eficiente de recursos e custos em ambientes de nuvem com aplicações web com IA
 - 18.8.4. Avaliação e comparação de fornecedores de serviços em nuvem para projetos web com Inteligência Artificial
 - 18.9. Criação de um Projeto com IA para Ambientes LAMP com a ajuda do ChatGPT
 - 18.9.1. Adaptação de projetos web baseados na pilha LAMP para incluir componentes de Inteligência Artificial
 - 18.9.2. Integração de bibliotecas e *frameworks* específicos para IA em ambientes LAMP
 - 18.9.3. Desenvolvimento de funcionalidades de IA que complementam a arquitetura LAMP tradicional
 - 18.9.4. Estratégias para a otimização e manutenção em projetos web com IA em ambientes LAMP
 - 18.10. Criação de um Projeto com IA para Ambientes MEVN usando o ChatGPT
 - 18.10.1. Integração de tecnologias e ferramentas da pilha MEVN com componentes de Inteligência Artificial
 - 18.10.2. Desenvolvimento de aplicações web modernas e escaláveis em ambientes MEVN com capacidades de IA
 - 18.10.3. Implementação de funcionalidades de processamento de dados e aprendizagem automática em projetos MEVN
 - 18.10.4. Estratégias para a melhoria do desempenho e da segurança em aplicações web com IA em ambientes MEVN
- Módulo 19. Aplicações móveis com IA**
- 19.1. Preparação do Ambiente de Trabalho para Desenvolvimento Móvel com IA
 - 19.1.1. Configuração de ambientes de desenvolvimento móvel para projetos com Inteligência Artificial
 - 19.1.2. Seleção e preparação de ferramentas específicas para o desenvolvimento de aplicações móveis com IA
 - 19.1.3. Integração de bibliotecas e *frameworks* de IA em ambientes de desenvolvimento móvel
 - 19.1.4. Configuração de emuladores e dispositivos reais para testar aplicações móveis com componentes de inteligência artificial
 - 19.2. Criação de um *Workspace* com GitHub Copilot
 - 19.2.1. Integração do GitHub Copilot em ambientes de desenvolvimento móvel
 - 19.2.2. Utilização eficaz do GitHub Copilot para a geração de código em projetos de IA
 - 19.2.3. Estratégias para colaboração de desenvolvedores ao usar o GitHub Copilot no *workspace*
 - 19.2.4. Boas práticas e limitações no uso do GitHub Copilot no desenvolvimento de aplicações móveis com IA

- 19.3. Configuração do Firebase
 - 19.3.1. Configuração inicial de um projeto no Firebase para o desenvolvimento móvel
 - 19.3.2. Integração do Firebase em aplicações móveis com funcionalidades de Inteligência Artificial
 - 19.3.3. Uso de serviços do Firebase como base de dados, autenticação e notificações em projetos com IA
 - 19.3.4. Estratégias para a gestão de dados e eventos em tempo real em aplicações móveis com Firebase
 - 19.4. Conceitos de Clean Architecture, DataSources, Repositories
 - 19.4.1. Princípios fundamentais de Clean Architecture no desenvolvimento móvel com IA
 - 19.4.2. Implementação de camadas de DataSources e Repositories com o GitHub Copilot
 - 19.4.3. Design e estruturação de componentes em projetos móveis com o GitHub Copilot
 - 19.4.4. Benefícios e desafios da implementação de *Clean Architecture* em aplicações móveis com IA
 - 19.5. Criação de Ecrã de Autenticação com GitHub Copilot
 - 19.5.1. Design e desenvolvimento de interfaces de utilizador para ecrãs de autenticação em aplicações móveis com IA
 - 19.5.2. Integração de serviços de autenticação com Firebase na ecrã de início de sessão
 - 19.5.3. Uso de técnicas de segurança e proteção de dados no ecrã de autenticação
 - 19.5.4. Personalização e adaptação da experiência de utilizador no ecrã de autenticação
 - 19.6. Criação de Dashboard e Navegação com GitHub Copilot
 - 19.6.1. Design e desenvolvimento de *Dashboards* com elementos de Inteligência Artificial
 - 19.6.2. Implementação de sistemas de navegação eficientes em aplicações móveis com IA
 - 19.6.3. Integração de funcionalidades de IA no *Dashboard* para melhorar a experiência do utilizador
 - 19.7. Criação de Ecrã com Listagem usando GitHub Copilot
 - 19.7.1. Desenvolvimento de interfaces de utilizador para ecrãs com listagens em aplicações móveis com IA
 - 19.7.2. Integração de algoritmos de recomendação e filtragem no ecrã de listagem
 - 19.7.3. Uso de padrões de design para a apresentação eficaz de dados na listagem
 - 19.7.4. Estratégias para o carregamento eficiente de dados em tempo real no ecrã com listagem
 - 19.8. Criação de Ecrã de Detalhes com GitHub Copilot
 - 19.8.1. Design e desenvolvimento de interfaces de utilizador detalhadas para a apresentação de informação específica
 - 19.8.2. Integração de funcionalidades de IA para enriquecer o ecrã de detalhes
 - 19.8.3. Implementação de interações e animações no ecrã de detalhes
 - 19.8.4. Estratégias para a otimização do desempenho no carregamento e visualização de detalhes em aplicações móveis com IA
 - 19.9. Criação de Ecrã de Configurações com GitHub Copilot
 - 19.9.1. Desenvolvimento de interfaces de utilizador para configuração e ajustes em aplicações móveis com IA
 - 19.9.2. Integração de ajustes personalizados relacionados com componentes de inteligência artificial
 - 19.9.3. Implementação de opções de personalização e preferências no ecrã de configurações
 - 19.9.4. Estratégias de usabilidade e clareza na apresentação das opções no ecrã de *settings*
 - 19.10. Criar ícones, *Splash* e Recursos Gráficos para a Sua App com IA
 - 19.10.1. Conceção e criação de ícones atrativos para representar a aplicação móvel com IA
 - 19.10.2. Desenvolvimento de ecrãs iniciais (*splash*) com elementos visuais impressionantes
 - 19.10.3. Seleção e adaptação de recursos gráficos para melhorar a estética da aplicação móvel
 - 19.10.4. Estratégias para a consistência e branding visual nos elementos gráficos da aplicação com IA
- Módulo 20. IA para QA Testing**
- 20.1. Ciclo de Vida dos *Testing*
 - 20.1.1. Descrição e compreensão do ciclo de vida de *testing* no desenvolvimento de software
 - 20.1.2. Fases do ciclo de vida de *testing* e a sua importância para a garantia da qualidade
 - 20.1.3. Integração da inteligência artificial em diferentes fases do ciclo de vida de *testing*
 - 20.1.4. Estratégias para a melhoria contínua do ciclo de vida de *testing* através da utilização da IA

- 20.2. Casos de Teste e Detecção de Bugs com a ajuda do ChatGPT
 - 20.2.1. Design e redação eficazes de casos de teste no contexto da QA *Testing*
 - 20.2.2. Identificação de bugs e erros durante a execução de casos de teste
 - 20.2.3. Aplicação de técnicas de detecção precoce de bugs através de análise estática
 - 20.2.4. Uso de ferramentas de inteligência artificial para identificação automática de bugs em casos de teste
- 20.3. Tipos de *Testing*
 - 20.3.1. Exploração de diferentes tipos de *testing* no domínio da QA
 - 20.3.2. Testes unitários, de integração, funcionais e de aceitação: características e aplicações
 - 20.3.3. Estratégias para a seleção e combinação adequada de tipos de testes em projetos com ChatGPT
 - 20.3.4. Adaptação de tipos de testes convencionais a projetos com ChatGPT
- 20.4. Criar um Plano de Testes
 - 20.4.1. Design e estruturação de um plano de testes integral
 - 20.4.2. Identificação de requisitos e cenários de teste em projetos com IA
 - 20.4.3. Estratégias para o planeamento de testes manuais e automatizados
 - 20.4.4. Avaliação e ajuste contínuo do plano de testes com base no desenvolvimento do projeto
- 20.5. Detecção e Notificação *Bugs* com IA
 - 20.5.1. Implementação de técnicas de detecção automática de bugs mediante algoritmos de aprendizagem automática
 - 20.5.2. Uso do ChatGPT para a análise dinâmica de código à procura de possíveis erros
 - 20.5.3. Estratégias para a geração automática de relatórios detalhados sobre bugs detetados usando o ChatGPT
 - 20.5.4. Colaboração eficaz entre equipas de desenvolvimento e QA na gestão de bugs identificados por IA
- 20.6. Criação de Testes Automatizados com IA
 - 20.6.1. Desenvolvimento de scripts de teste automatizados para projetos usando o ChatGPT
 - 20.6.2. Integração de ferramentas de automatização de testes baseadas em IA
 - 20.6.3. Uso do ChatGPT para a geração dinâmica de casos de teste automatizados
 - 20.6.4. Estratégias para a execução eficiente e manutenção de testes automatizados em projetos com IA
- 20.7. *API Testing*
 - 20.7.1. Conceitos fundamentais de *API testing* e a sua importância em QA
 - 20.7.2. Desenvolvimento de testes para a verificação de APIs em ambientes usando o ChatGPT
 - 20.7.3. Estratégias para a validação de dados e resultados em testes de APIs com o ChatGPT
 - 20.7.4. Uso de ferramentas específicas para o *testing* de APIs em projetos com inteligência artificial
- 20.8. Ferramentas de IA para a *web Testing*
 - 20.8.1. Exploração de ferramentas de inteligência artificial para automatização de testes em ambientes web
 - 20.8.2. Integração de tecnologias de reconhecimento de elementos e de análise visual na *web testing*
 - 20.8.3. Estratégias para a deteção automática de alterações e problemas de desempenho em aplicações web utilizando o ChatGPT
 - 20.8.4. Avaliação de ferramentas específicas para melhorar a eficiência nos *web testing* com IA
- 20.9. *Mobile Testing* Através da IA
 - 20.9.1. Desenvolvimento de estratégias de *testing* para aplicações móveis com componentes de inteligência artificial
 - 20.9.2. Integração de ferramentas de *testing* específicas para plataformas móveis baseadas em IA
 - 20.9.3. Uso do ChatGPT para a deteção de problemas no desempenho de aplicações móveis
 - 20.9.4. Estratégias para a validação de interfaces e funções específicas de aplicações móveis através de IA
- 20.10. Ferramentas de QA com IA
 - 20.10.1. Exploração de ferramentas e plataformas de QA que incorporam funcionalidades de Inteligência Artificial
 - 20.10.2. Avaliação de ferramentas para a gestão e execução eficiente de testes em projetos com IA
 - 20.10.3. Uso do ChatGPT para a geração e otimização de casos de teste
 - 20.10.4. Estratégias para a seleção e adoção eficaz de ferramentas de QA com capacidades de IA

06

Metodologia de estudo

A TECH é a primeira universidade do mundo a unir a metodologia dos **case studies** com o **Relearning**, um sistema de aprendizado 100% online baseado na repetição guiada.

Essa estratégia de ensino inovadora foi projetada para oferecer aos profissionais a oportunidade de atualizar conhecimentos e desenvolver habilidades de forma intensiva e rigorosa. Um modelo de aprendizagem que coloca o aluno no centro do processo acadêmico e lhe dá o papel principal, adaptando-se às suas necessidades e deixando de lado as metodologias mais convencionais.



“

A TECH prepara você para enfrentar novos desafios em ambientes incertos e alcançar o sucesso em sua carreira”

Estudo de Caso para contextualizar todo o conteúdo

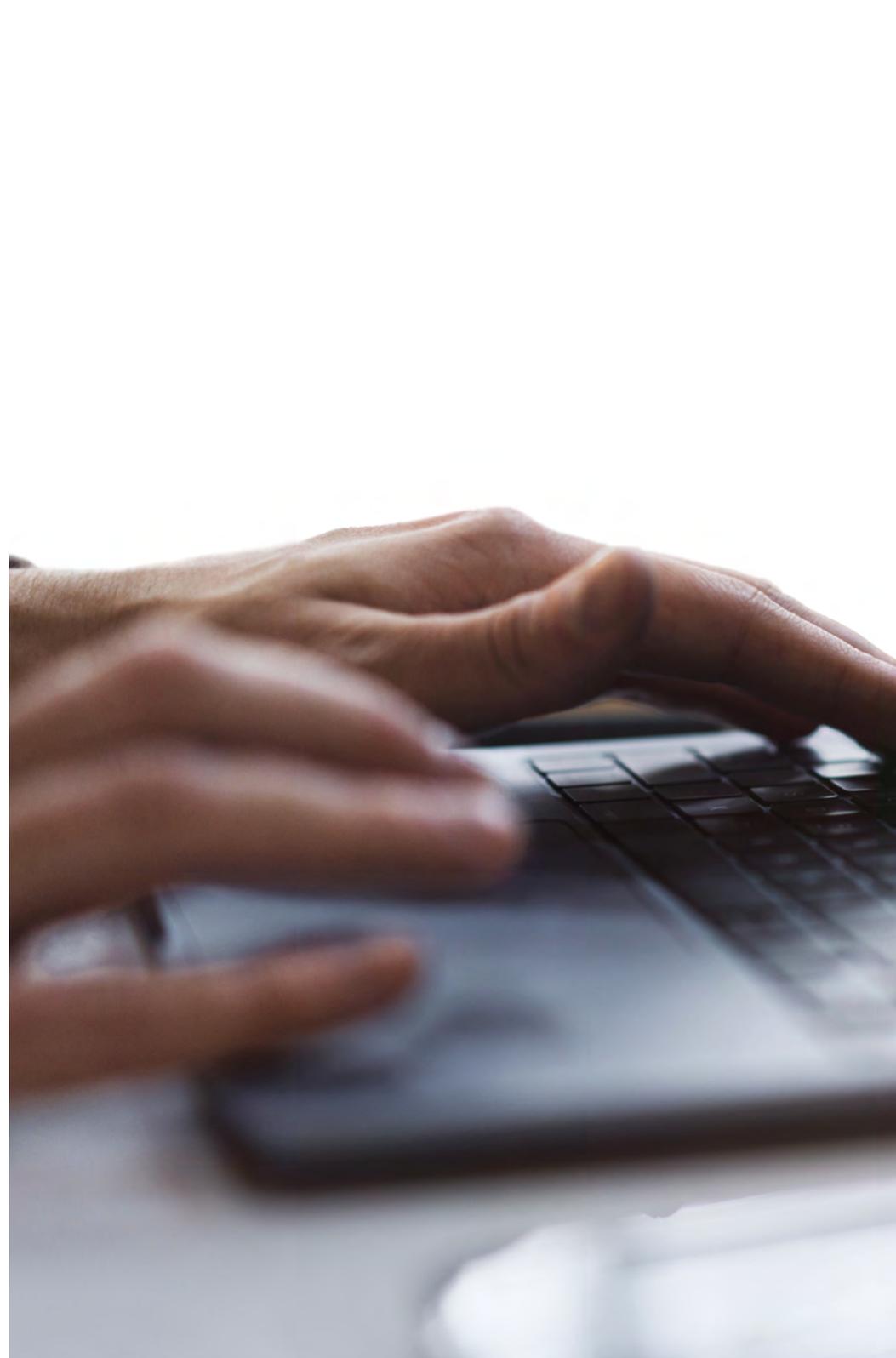
O aluno: a prioridade de todos os programas da TECH

Na metodologia de estudo da TECH, o aluno é o protagonista absoluto. As ferramentas pedagógicas de cada programa foram selecionadas levando-se em conta as demandas de tempo, disponibilidade e rigor acadêmico que, atualmente, os alunos, bem como os empregos mais competitivos do mercado, exigem.

Com o modelo educacional assíncrono da TECH, é o aluno quem escolhe quanto tempo passa estudando, como decide estabelecer suas rotinas e tudo isso no conforto do dispositivo eletrônico de sua escolha. O aluno não precisa assistir às aulas presenciais, que muitas vezes não poderá comparecer. As atividades de aprendizado serão realizadas de acordo com sua conveniência. O aluno sempre poderá decidir quando e de onde estudar.

“

*Na TECH, o aluno NÃO terá aulas ao vivo
(das quais poderá nunca participar)”*



Os programas de ensino mais abrangentes do mundo

A TECH se caracteriza por oferecer os programas acadêmicos mais completos no ambiente universitário. Essa abrangência é obtida por meio da criação de programas de estudo que cobrem não apenas o conhecimento essencial, mas também as últimas inovações em cada área.

Por serem constantemente atualizados, esses programas permitem que os alunos acompanhem as mudanças do mercado e adquiram as habilidades mais valorizadas pelos empregadores. Dessa forma, os alunos da TECH recebem uma preparação abrangente que lhes dá uma vantagem competitiva significativa para avançar em suas carreiras.

Além disso, eles podem fazer isso de qualquer dispositivo, PC, tablet ou smartphone.

“

O modelo da TECH é assíncrono, portanto, você poderá estudar com seu PC, tablet ou smartphone onde quiser, quando quiser e pelo tempo que quiser”

Case studies ou Método de caso

O método de casos tem sido o sistema de aprendizado mais amplamente utilizado pelas melhores escolas de negócios do mundo. Desenvolvido em 1912 para que os estudantes de direito não aprendessem a lei apenas com base no conteúdo teórico, sua função também era apresentar a eles situações complexas da vida real. Assim, eles poderiam tomar decisões informadas e fazer julgamentos de valor sobre como resolvê-los. Em 1924 foi estabelecido como o método de ensino padrão em Harvard.

Com esse modelo de ensino, é o próprio aluno que desenvolve sua competência profissional por meio de estratégias como o *Learning by doing* ou o *Design Thinking*, usados por outras instituições renomadas, como Yale ou Stanford.

Esse método orientado para a ação será aplicado em toda a trajetória acadêmica do aluno com a TECH. Dessa forma, o aluno será confrontado com várias situações da vida real e terá de integrar conhecimentos, pesquisar, argumentar e defender suas ideias e decisões. A premissa era responder à pergunta sobre como eles agiriam diante de eventos específicos de complexidade em seu trabalho diário.



Método Relearning

Na TECH os *case studies* são alimentados pelo melhor método de ensino 100% online: o *Relearning*.

Esse método rompe com as técnicas tradicionais de ensino para colocar o aluno no centro da equação, fornecendo o melhor conteúdo em diferentes formatos. Dessa forma, consegue revisar e reiterar os principais conceitos de cada matéria e aprender a aplicá-los em um ambiente real.

Na mesma linha, e de acordo com várias pesquisas científicas, a repetição é a melhor maneira de aprender. Portanto, a TECH oferece entre 8 e 16 repetições de cada conceito-chave dentro da mesma lição, apresentadas de uma forma diferente, a fim de garantir que o conhecimento seja totalmente incorporado durante o processo de estudo.

O Relearning permitirá uma aprendizagem com menos esforço e mais desempenho, fazendo com que você se envolva mais em sua especialização, desenvolvendo seu espírito crítico e sua capacidade de defender argumentos e contrastar opiniões: uma equação de sucesso.



Um Campus Virtual 100% online com os melhores recursos didáticos

Para aplicar sua metodologia de forma eficaz, a TECH se concentra em fornecer aos alunos materiais didáticos em diferentes formatos: textos, vídeos interativos, ilustrações e mapas de conhecimento, entre outros. Todos eles são projetados por professores qualificados que concentram seu trabalho na combinação de casos reais com a resolução de situações complexas por meio de simulação, o estudo de contextos aplicados a cada carreira profissional e o aprendizado baseado na repetição, por meio de áudios, apresentações, animações, imagens etc.

As evidências científicas mais recentes no campo da neurociência apontam para importância de levar em conta o local e o contexto em que o conteúdo é acessado antes de iniciar um novo processo de aprendizagem. A capacidade de ajustar essas variáveis de forma personalizada ajuda as pessoas a lembrar e armazenar o conhecimento no hipocampo para retenção a longo prazo. Trata-se de um modelo chamado *Neurocognitive context-dependent e-learning* que é aplicado conscientemente nesse curso universitário.

Por outro lado, também para favorecer ao máximo o contato entre mentor e mentorado, é oferecida uma ampla variedade de possibilidades de comunicação, tanto em tempo real quanto em diferido (mensagens internas, fóruns de discussão, serviço telefônico, contato por e-mail com a secretaria técnica, bate-papo, videoconferência etc.).

Da mesma forma, esse Campus Virtual muito completo permitirá que os alunos da TECH organizem seus horários de estudo de acordo com sua disponibilidade pessoal ou obrigações de trabalho. Dessa forma, eles terão um controle global dos conteúdos acadêmicos e de suas ferramentas didáticas, em função de sua atualização profissional acelerada.



O modo de estudo online deste programa permitirá que você organize seu tempo e ritmo de aprendizado, adaptando-o à sua agenda”

A eficácia do método é justificada por quatro conquistas fundamentais:

1. Os alunos que seguem este método não só assimilam os conceitos, mas também desenvolvem a capacidade intelectual através de exercícios de avaliação de situações reais e de aplicação de conhecimentos.
2. A aprendizagem se consolida nas habilidades práticas, permitindo ao aluno integrar melhor o conhecimento à prática clínica.
3. A assimilação de ideias e conceitos se torna mais fácil e eficiente, graças à abordagem de situações decorrentes da realidade.
4. A sensação de eficiência do esforço investido se torna um estímulo muito importante para os alunos, o que se traduz em um maior interesse pela aprendizagem e um aumento no tempo dedicado ao curso.



A metodologia universitária mais bem avaliada por seus alunos

Os resultados desse modelo acadêmico inovador podem ser vistos nos níveis gerais de satisfação dos alunos da TECH.

A avaliação dos alunos sobre a qualidade do ensino, a qualidade dos materiais, a estrutura e os objetivos do curso é excelente. Não é de surpreender que a instituição tenha se tornado a universidade mais bem avaliada por seus alunos na plataforma de avaliação Trustpilot, com uma pontuação de 4,9 de 5.

Acesse o conteúdo do estudo de qualquer dispositivo com conexão à Internet (computador, tablet, smartphone) graças ao fato da TECH estar na vanguarda da tecnologia e do ensino.

Você poderá aprender com as vantagens do acesso a ambientes de aprendizagem simulados e com a abordagem de aprendizagem por observação, ou seja, aprender com um especialista.

Assim, os melhores materiais educacionais, cuidadosamente preparados, estarão disponíveis neste programa:



Material de estudo

O conteúdo didático foi elaborado especialmente para este curso pelos especialistas que irão ministrá-lo, o que permite que o desenvolvimento didático seja realmente específico e concreto.

Posteriormente, esse conteúdo é adaptado ao formato audiovisual, para criar o método de trabalho online, com as técnicas mais recentes que nos permitem lhe oferecer a melhor qualidade em cada uma das peças que colocaremos a seu serviço.



Práticas de aptidões e competências

Serão realizadas atividades para desenvolver as habilidades e competências específicas em cada área temática. Práticas e dinâmicas para adquirir e desenvolver as competências e habilidades que um especialista precisa desenvolver no âmbito da globalização.



Resumos interativos

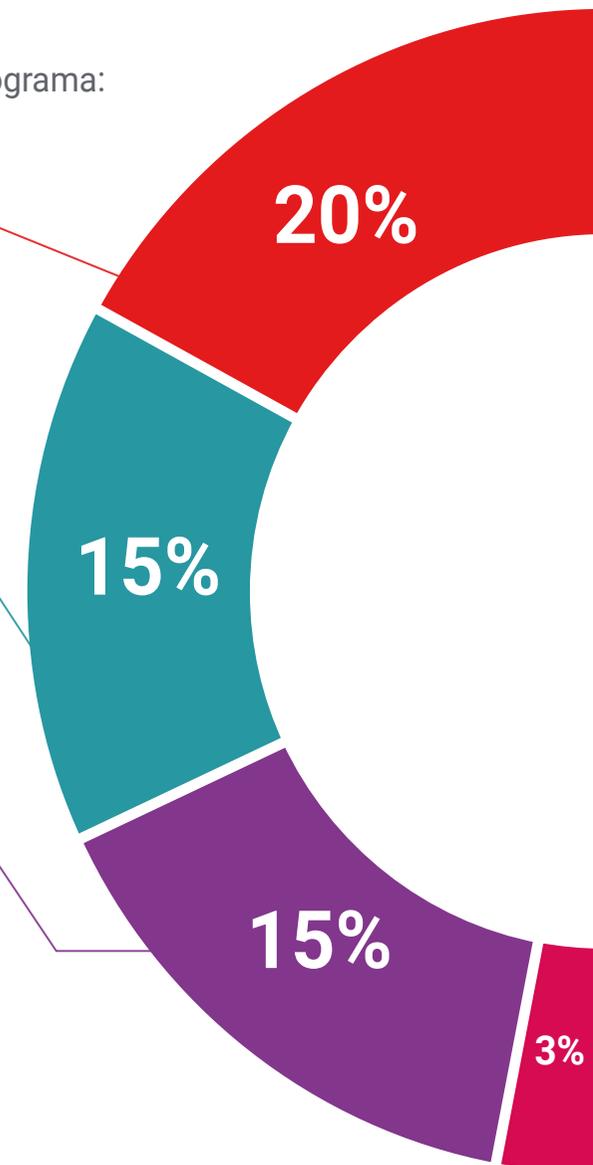
Apresentamos os conteúdos de forma atraente e dinâmica em pilulas multimídia que incluem áudio, vídeos, imagens, diagramas e mapas conceituais com o objetivo de reforçar o conhecimento.

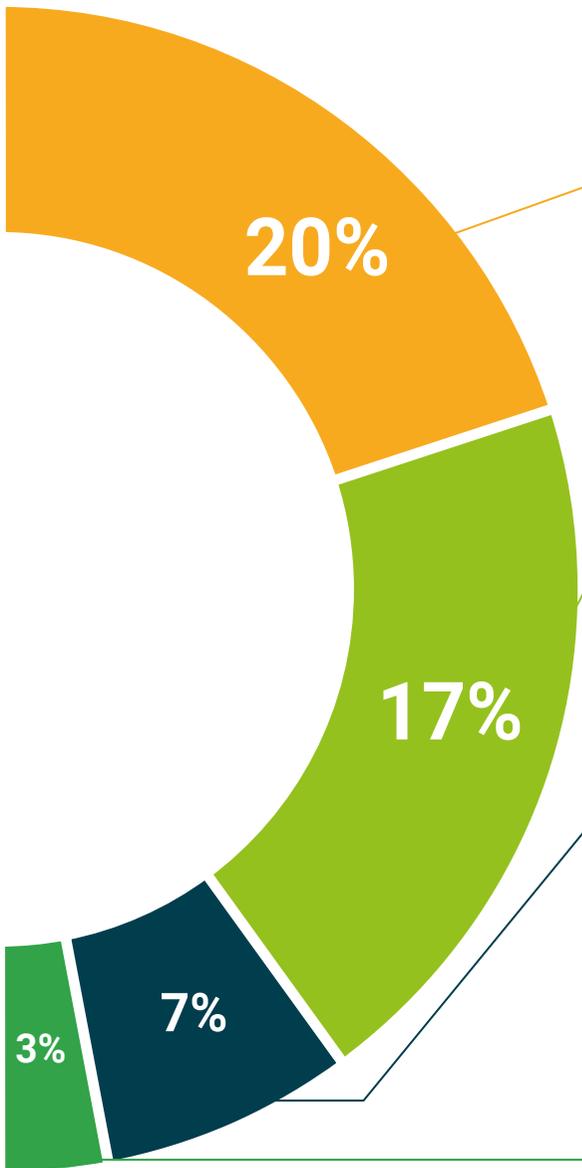
Este sistema exclusivo de capacitação por meio da apresentação de conteúdo multimídia foi premiado pela Microsoft como "Caso de sucesso na Europa"



Leituras complementares

Artigos recentes, documentos científicos, guias internacionais, entre outros. Na biblioteca virtual do estudante você terá acesso a tudo o que for necessário para completar sua capacitação.





Case Studies

Você concluirá uma seleção dos melhores *case studies* da disciplina. Casos apresentados, analisados e orientados pelos melhores especialistas no cenário internacional.



Testing & Retesting

Avaliamos e reavaliamos periodicamente seus conhecimentos ao longo de todo o programa. Fazemos isso em 3 dos 4 níveis da Pirâmide de Miller.



Masterclasses

Há evidências científicas sobre a utilidade da observação de terceiros especialistas. O *Learning from an expert* fortalece o conhecimento e a memória, e aumenta nossa confiança para tomar decisões difíceis no futuro.



Guias rápidos de ação

A TECH oferece o conteúdo mais relevante do curso em formato de fichas de trabalho ou guias rápidos de ação. Uma forma sintetizada, prática e eficaz de ajudar os alunos a progredirem na aprendizagem.



07

Certificação

O Mestrado Próprio em Inteligência Artificial na Programação garante, além da formação mais rigorosa e atualizada, o acesso a um certificado de Curso emitido pela TECH Universidade Tecnológica.



“

*Conclua este programa de estudos
com sucesso e receba seu certificado
sem sair de casa e sem burocracias”*

Este **Mestrado Próprio em Inteligência Artificial na Programação** conta com o conteúdo educacional mais completo e atualizado do mercado.

Uma vez aprovadas as avaliações, o aluno receberá por correio, com aviso de receção, o certificado* correspondente ao título de **Mestrado Próprio** emitido pela **TECH Universidade Tecnológica**.

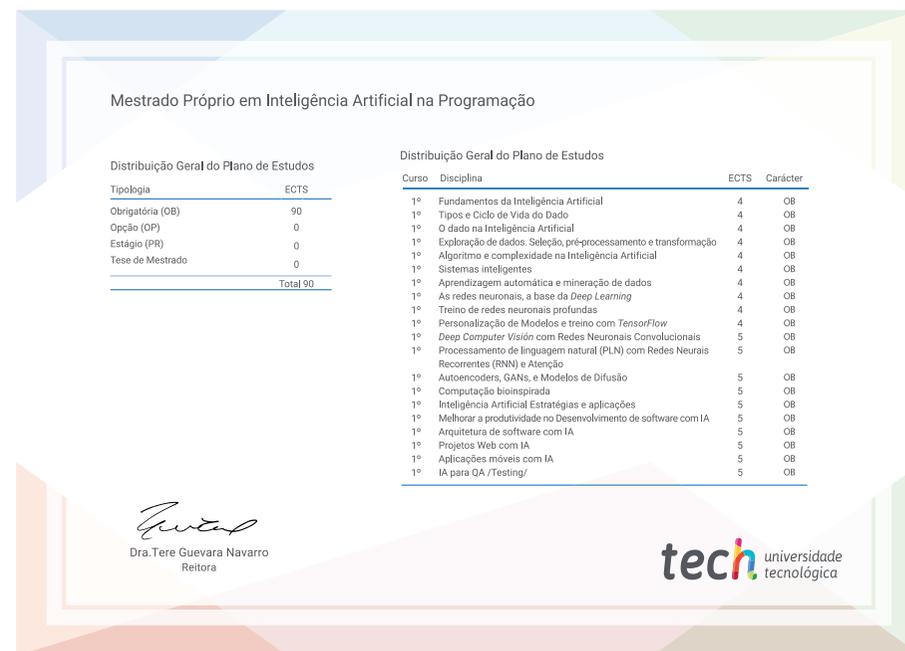
O certificado emitido pela **TECH Universidade Tecnológica** expressará a qualificação obtida no Mestrado Próprio, atendendo aos requisitos normalmente exigidos pelas bolsas de emprego, concursos públicos e avaliação de carreiras profissionais.

Certificação: **Mestrado Próprio em Inteligência Artificial na Programação**

Modalidade: **online**

Duração: **12 meses**

ECTS: **90**



*Apostila de Haia: Caso o aluno solicite que o seu certificado seja apostilado, a TECH EDUCATION providenciará a obtenção do mesmo a um custo adicional.

futuro
saúde confiança pessoas
informação orientadores
educação certificação ensino
garantia aprendizagem
instituições tecnologia
comunidade comunidade
atenção personalizada
conhecimento inovação
presente qualidade
desenvolvimento sustentabilidade



Mestrado Próprio Inteligência Artificial na Programação

- » Modalidade: online
- » Duração: 6 meses
- » Certificação: TECH Universidade Tecnológica
- » Acreditação: 90 ECTS
- » Horário: a tua scelta
- » Exames: online

Mestrado Próprio

Inteligência Artificial na Programação