

Máster de Formación Permanente Inteligencia Artificial en la Programación



Máster de Formación Permanente Inteligencia Artificial en la Programación

- » Modalidad: **online**
- » Duración: **12 meses**
- » Titulación: **TECH Universidad**
- » Acreditación: **90 ECTS**
- » Horario: **a tu ritmo**
- » Exámenes: **online**

Acceso web: www.techtitute.com/informatica/master/master-inteligencia-artificial-programacion

Índice

01

Presentación del programa

pág. 4

02

¿Por qué estudiar en TECH?

pág. 8

03

Plan de estudios

pág. 12

04

Objetivos docentes

pág. 36

05

Salidas profesionales

pág. 44

06

Licencias de software incluidas

pág. 48

07

Metodología de estudio

pág. 52

08

Cuadro docente

pág. 62

09

Titulación

pág. 66

01

Presentación del programa

La incorporación de la Inteligencia Artificial en la Programación ha transformado radicalmente la manera en que se diseñan optimiza y ejecutan soluciones digitales. De hecho, la IA se ha consolidado como una herramienta clave en el desarrollo de software. Según un informe de la Comisión Europea, el 61% de las empresas en el sector tecnológico ya utilizan aplicaciones basadas en IA para mejorar su eficiencia y competitividad. En este contexto, surge esta oportunidad académica de TECH como respuesta a la creciente demanda de profesionales capaces de integrar estos avances en proyectos reales. A su vez, mediante una metodología estructurada, que combina material didáctico especializado y una modalidad 100% online, se garantizará una capacitación adaptada a los desafíos actuales.





Gracias a este Máster de Formación Permanente 100% online, dominarás el uso de la Inteligencia Artificial aplicada a la Programación para optimizar procesos y resolver desafíos tecnológicos”

Actualmente, la incorporación de sistemas inteligentes en el desarrollo de *software* ha transformado radicalmente la manera en la que ejecutan procesos informáticos. En este contexto, la Inteligencia Artificial se ha convertido en una herramienta esencial para automatizar tareas complejas, detectar errores en tiempo real y generar soluciones más eficientes. Por lo tanto, su impacto abarca desde la creación de algoritmos adaptativos hasta la personalización de servicios digitales, lo cual resulta especialmente valioso en industrias que requieren respuestas ágiles y precisas.

Consciente de este panorama, TECH Universidad desarrolla un enfoque académico que combina los fundamentos teóricos con aplicaciones prácticas de vanguardia. A su vez, el estudio de la Inteligencia Artificial se abordará desde una perspectiva estratégica, articulándose con el análisis de grandes volúmenes de datos en entornos *big data*. Asimismo, se profundizará en la construcción de algoritmos que respondan a las exigencias actuales del sector tecnológico, brindando una comprensión sólida de sus mecanismos y aplicaciones.

Posteriormente, al completar este programa universitario, los profesionales ampliarán significativamente su capacidad de diseñar soluciones inteligentes y optimizar procesos mediante el uso de modelos algorítmicos avanzados. Asimismo, estarán en condiciones de aplicar técnicas analíticas para extraer valor de grandes bases de datos y crear sistemas predictivos con impacto real en entornos productivos. De hecho, esta especialización no solo fortalecerá habilidades técnicas, sino que también impulsará la toma de decisiones estratégicas con base en información procesada de manera inteligente y automatizada.

Finalmente, la metodología impulsada por TECH responde a las necesidades actuales de quienes buscan una experiencia académica flexible y de alto nivel. Gracias a su enfoque digital, el acceso es continuo, los 7 días de la semana y desde cualquier lugar con conexión a internet. No obstante, este modelo se apoya en el sistema *Relearning*, el cual refuerza la capacitación mediante la repetición estratégica de conceptos clave, optimizando la retención y favoreciendo una comprensión profunda.

Este **Máster de Formación Permanente en Inteligencia Artificial en la Programación** contiene el programa universitario más completo y actualizado del mercado.

Sus características más destacadas son:

- ♦ El desarrollo de casos prácticos presentados por expertos en Inteligencia Artificial en la Programación
- ♦ Los contenidos gráficos, esquemáticos y eminentemente prácticos con los que están concebidos recogen una información científica y práctica sobre aquellas disciplinas indispensables para el ejercicio profesional
- ♦ Los ejercicios prácticos donde realizar el proceso de autoevaluación para mejorar el aprendizaje
- ♦ Su especial hincapié en metodologías innovadoras
- ♦ Las lecciones teóricas, preguntas al experto, foros de discusión de temas controvertidos y trabajos de reflexión individual
- ♦ La disponibilidad de acceso a los contenidos desde cualquier dispositivo fijo o portátil con conexión a internet



Adquirirás un conocimiento integral sobre la incorporación de sistemas inteligentes en el desarrollo de software, lo que garantizará soluciones avanzadas”

“

Te sumergirás en los aspectos fundamentales de la arquitectura del software, gracias a los recursos multimedia más innovadores del mercado”

Incluye en su cuadro docente a profesionales pertenecientes al ámbito de la Inteligencia Artificial en la Programación, que vierten en este programa la experiencia de su trabajo, además de reconocidos especialistas de sociedades de referencia y universidades de prestigio.

Su contenido multimedia, elaborado con la última tecnología educativa, permitirá al profesional un aprendizaje situado y contextualizado, es decir, un entorno simulado que proporcionará un estudio inmersivo programado para entrenarse ante situaciones reales.

El diseño de este programa se centra en el Aprendizaje Basado en Problemas, mediante el cual el alumno deberá tratar de resolver las distintas situaciones de práctica profesional que se le planteen a lo largo del curso académico. Para ello, el profesional contará con la ayuda de un novedoso sistema de vídeo interactivo realizado por reconocidos expertos.

Profundizarás en la integración de elementos de IA en Visual Studio Code, a través de un programa universitario de alto nivel.

Con el sistema Relearning que emplea TECH reducirás las largas horas de estudio y memorización.



02

¿Por qué estudiar en TECH?

TECH es la mayor Universidad digital del mundo. Con un impresionante catálogo de más de 14.000 programas universitarios, disponibles en 11 idiomas, se posiciona como líder en empleabilidad, con una tasa de inserción laboral del 99%. Además, cuenta con un enorme claustro de más de 6.000 profesores de máximo prestigio internacional.



“

Estudia en la mayor universidad digital del mundo y asegura tu éxito profesional. El futuro empieza en TECH”

La mejor universidad online del mundo según FORBES

La prestigiosa revista Forbes, especializada en negocios y finanzas, ha destacado a TECH como «la mejor universidad online del mundo». Así lo han hecho constar recientemente en un artículo de su edición digital en el que se hacen eco del caso de éxito de esta institución, «gracias a la oferta académica que ofrece, la selección de su personal docente, y un método de aprendizaje innovador orientado a formar a los profesionales del futuro».

Forbes
Mejor universidad
online del mundo

Plan
de estudios
más completo

Los planes de estudio más completos del panorama universitario

TECH ofrece los planes de estudio más completos del panorama universitario, con temarios que abarcan conceptos fundamentales y, al mismo tiempo, los principales avances científicos en sus áreas científicas específicas. Asimismo, estos programas son actualizados continuamente para garantizar al alumnado la vanguardia académica y las competencias profesionales más demandadas. De esta forma, los títulos de la universidad proporcionan a sus egresados una significativa ventaja para impulsar sus carreras hacia el éxito.

El mejor claustro docente top internacional

El claustro docente de TECH está integrado por más de 6.000 profesores de máximo prestigio internacional. Catedráticos, investigadores y altos ejecutivos de multinacionales, entre los cuales se destacan Isaiah Covington, entrenador de rendimiento de los Boston Celtics; Magda Romanska, investigadora principal de MetaLAB de Harvard; Ignacio Wistuba, presidente del departamento de patología molecular traslacional del MD Anderson Cancer Center; o D.W Pine, director creativo de la revista TIME, entre otros.

Profesorado
TOP
Internacional



La metodología
más eficaz

Un método de aprendizaje único

TECH es la primera universidad que emplea el *Relearning* en todas sus titulaciones. Se trata de la mejor metodología de aprendizaje online, acreditada con certificaciones internacionales de calidad docente, dispuestas por agencias educativas de prestigio. Además, este disruptivo modelo académico se complementa con el "Método del Caso", configurando así una estrategia de docencia online única. También en ella se implementan recursos didácticos innovadores entre los que destacan vídeos en detalle, infografías y resúmenes interactivos.

La mayor universidad digital del mundo

TECH es la mayor universidad digital del mundo. Somos la mayor institución educativa, con el mejor y más amplio catálogo educativo digital, cien por cien online y abarcando la gran mayoría de áreas de conocimiento. Ofrecemos el mayor número de titulaciones propias, titulaciones oficiales de posgrado y de grado universitario del mundo. En total, más de 14.000 títulos universitarios, en once idiomas distintos, que nos convierten en la mayor institución educativa del mundo.

nº1
Mundial
Mayor universidad
online del mundo

La universidad online oficial de la NBA

TECH es la universidad online oficial de la NBA. Gracias a un acuerdo con la mayor liga de baloncesto, ofrece a sus alumnos programas universitarios exclusivos, así como una gran variedad de recursos educativos centrados en el negocio de la liga y otras áreas de la industria del deporte. Cada programa tiene un currículo de diseño único y cuenta con oradores invitados de excepción: profesionales con una distinguida trayectoria deportiva que ofrecerán su experiencia en los temas más relevantes.

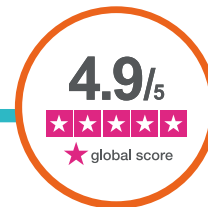
Líderes en empleabilidad

TECH ha conseguido convertirse en la universidad líder en empleabilidad. El 99% de sus alumnos obtienen trabajo en el campo académico que ha estudiado, antes de completar un año luego de finalizar cualquiera de los programas de la universidad. Una cifra similar consigue mejorar su carrera profesional de forma inmediata. Todo ello gracias a una metodología de estudio que basa su eficacia en la adquisición de competencias prácticas, totalmente necesarias para el desarrollo profesional.



Google Partner Premier

El gigante tecnológico norteamericano ha otorgado a TECH la insignia Google Partner Premier. Este galardón, solo al alcance del 3% de las empresas del mundo, pone en valor la experiencia eficaz, flexible y adaptada que esta universidad proporciona al alumno. El reconocimiento no solo acredita el máximo rigor, rendimiento e inversión en las infraestructuras digitales de TECH, sino que también sitúa a esta universidad como una de las compañías tecnológicas más punteras del mundo.



La universidad mejor valorada por sus alumnos

Los alumnos han posicionado a TECH como la universidad mejor valorada del mundo en los principales portales de opinión, destacando su calificación más alta de 4,9 sobre 5, obtenida a partir de más de 1.000 reseñas. Estos resultados consolidan a TECH como la institución universitaria de referencia a nivel internacional, reflejando la excelencia y el impacto positivo de su modelo educativo.



03

Plan de estudios

El completísimo itinerario que complementa este programa universitario abordará conceptos clave de la Inteligencia Artificial en la Programación, destacando técnicas avanzadas esenciales para el desarrollo de sistemas inteligentes. A lo largo de la titulación universitaria, el alumnado explorará redes neuronales convolucionales, así como la notación asintótica, crucial para evaluar la eficiencia de los algoritmos en entornos de gran escala. Además, el algoritmo alfa - beta será clave en la optimización de procesos de decisión en sistemas predictivos. Finalmente, la integración de estos temas permitirá a los profesionales afrontar desafíos tecnológicos con soluciones más eficientes en el desarrollo de software.



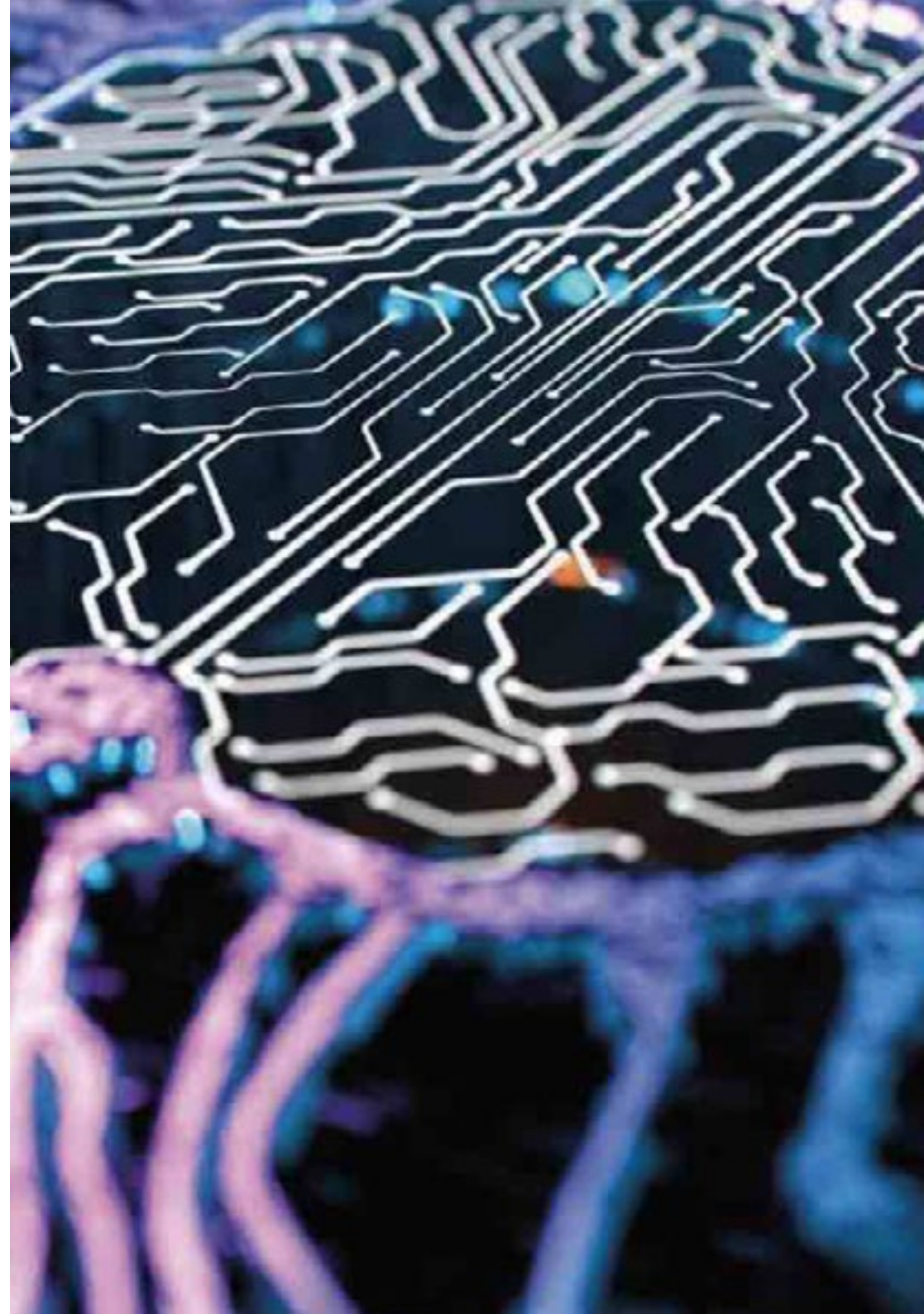


“

Gracias a esta efectiva preparación académica, fortalecerás tu comprensión y aplicación del algoritmo alfa - beta”

Módulo 1. Fundamentos de la Inteligencia Artificial

- 1.1. Historia de la Inteligencia Artificial
 - 1.1.1. ¿Cuándo se empieza a hablar de Inteligencia Artificial?
 - 1.1.2. Referentes en el cine
 - 1.1.3. Importancia de la Inteligencia Artificial
 - 1.1.4. Tecnologías que habilitan y dan soporte a la Inteligencia Artificial
- 1.2. La Inteligencia Artificial en juegos
 - 1.2.1. Terminología en los juegos
 - 1.2.1.1. Jugador
 - 1.2.1.2. Estados
 - 1.2.1.3. Acciones o movimientos
 - 1.2.1.4. Reglas
 - 1.2.1.5. Terminales
 - 1.2.1.6. Función de utilidad o recompensa
 - 1.2.1.7. Estrategia
 - 1.2.2. Clasificación de los juegos en IA
 - 1.2.2.1. Según la información disponible
 - 1.2.2.2. Según la existencia de azar
 - 1.2.2.3. Según la cantidad de jugadores
 - 1.2.2.4. Según el resultado
 - 1.2.2.5. Según la secuencia de movimientos
 - 1.2.3. Teoría de juegos
 - 1.2.4. Juegos de dos jugadores
 - 1.2.4.1. Juegos deterministas
 - 1.2.4.2. De información perfecta
 - 1.2.4.3. Por turnos
 - 1.2.4.4. Juegos de dos jugadores
 - 1.2.4.5. Ejemplos clásicos
 - 1.2.5. Árboles de juego
 - 1.2.5.1. Estructura del árbol
 - 1.2.5.2. Profundidad del árbol
 - 1.2.5.3. Niveles MAX y MIN
 - 1.2.5.4. Función de evaluación



- 1.3. Redes de neuronas
 - 1.3.1. Fundamentos biológicos
 - 1.3.2. Modelo computacional
 - 1.3.3. Redes de neuronas supervisadas y no supervisadas
 - 1.3.3.1. Redes supervisadas
 - 1.3.3.2. Redes no supervisadas
 - 1.3.4. Perceptrón simple
 - 1.3.5. Perceptrón multicapa
- 1.4. Algoritmos genéticos
 - 1.4.1. Historia
 - 1.4.2. Base biológica
 - 1.4.3. Codificación de problemas
 - 1.4.4. Generación de la población inicial
 - 1.4.5. Algoritmo principal y operadores genéticos
 - 1.4.5.1. Estructura general del algoritmo
 - 1.4.5.2. Operadores genéticos
 - 1.4.5.3. Representación del pseudocódigo del algoritmo
 - 1.4.6. Evaluación de individuos (Fitness)
 - 1.4.7. Aplicaciones
 - 1.4.7.1. Ingeniería y optimización
 - 1.4.7.2. Informática y ciencia de datos
 - 1.4.7.3. Economía, finanzas y logística
 - 1.4.7.4. Biología computacional
- 1.5. Introducción a los agentes inteligentes: tipos y características
 - 1.5.1. Vocabularios
 - 1.5.2. Taxonomías
 - 1.5.3. Tesoros
 - 1.5.4. Ontologías
- 1.6. Web semántica y representación ontológica
 - 1.6.1. Introducción: Web Semántica
 - 1.6.1.1. Principios fundamentales
 - 1.6.1.2. Arquitectura y niveles de la Web Semántica
 - 1.6.1.3. Ventajas de la Web Semántica
 - 1.6.2. RDF y RDFS: Descripción de recursos
 - 1.6.2.1. RDF Schema (RDFS)
 - 1.6.2.2. Importancia de RDF y RDFS
 - 1.6.3. OWL: Lenguaje de Ontologías
 - 1.6.3.1. Niveles de OWL
 - 1.6.3.2. Componentes de OWL
 - 1.6.3.3. Razonamiento e inferencia
 - 1.6.4. SPARQL y consultas semánticas
 - 1.6.4.1. Estructura básica de una consulta SPARQL
 - 1.6.4.2. Almacenamiento y ejecución de consultas
 - 1.6.5. Aplicaciones de la Web Semántica
 - 1.6.5.1. Gobierno y datos abiertos
 - 1.6.5.2. Ciencias de la salud y biología
 - 1.6.5.3. Industria y comercio electrónico
 - 1.6.5.4. Educación e investigación
 - 1.6.5.5. Web de datos enlazados (Linked Data)
- 1.7. Sistemas expertos y DSS
 - 1.7.1. Sistemas expertos
 - 1.7.1.1. Características ampliadas de los sistemas expertos
 - 1.7.1.2. Componentes principales (ampliados)
 - 1.7.1.3. Técnicas de representación del conocimiento
 - 1.7.2. Mecanismos de inferencia
 - 1.7.2.1. Encadenamiento hacia adelante
 - 1.7.2.2. Encadenamiento hacia atrás
 - 1.7.2.3. Ciclo de razonamiento ampliado
 - 1.7.3. Sistemas de soporte a la decisión (DSS)
 - 1.7.3.1. Componentes ampliados de un DSS
 - 1.7.3.2. Tipos de DSS ampliados
 - 1.7.4. Aplicaciones ampliadas
- 1.8. Chatbots y asistentes virtuales
 - 1.8.1. Tipos de asistentes
 - 1.8.2. Asistentes de voz
 - 1.8.3. Asistentes de texto
 - 1.8.4. Partes fundamentales, para el desarrollo de un asistente

- 1.8.5. *Intents*
- 1.8.6. Entidades
- 1.8.7. Flujo de dialogo
- 1.8.8. Integraciones: web, slack, Whatsapp, Facebook
- 1.8.9. Algunas herramientas para el desarrollo de asistentes: dialog Flow, Watson Assitant
- 1.8.10. Bibliografía
- 1.9. Estrategias de implantación de IA
 - 1.9.1. Un plan estratégico como pilar de implantación responsable
 - 1.9.2. Claves para implantar la IA
 - 1.9.3. Guía de implantación de IA
 - 1.9.4. Tips de ayuda en la implantación de IA
- 1.10. El futuro de la Inteligencia Artificial
 - 1.10.1. Reflexiones
 - 1.10.2. Ejemplos
 - 1.10.3. Posibilidades

Módulo 2. El Dato en la Inteligencia Artificial

- 2.1. Ciencia de datos y toma de decisiones automatizadas
 - 2.1.1. Ciencia de datos
 - 2.1.1.1. Lenguajes de programación para el análisis de datos
 - 2.1.1.2. Lenguajes más utilizados
 - 2.1.2. Herramientas avanzadas para el científico de datos
 - 2.1.2.1. Evolución de Python
 - 2.1.2.2. Características de Python
 - 2.1.2.3. Instalación de Python
 - 2.1.2.4. Otras herramientas: Anaconda
- 2.2. Datos, información y conocimiento
 - 2.2.1. Datos, información y conocimiento
 - 2.2.2. Tipos de datos
 - 2.2.3. Tipos de datos básicos
 - 2.2.4. Tipos de datos complejos
 - 2.2.5. Operaciones por tipo
 - 2.2.6. Estructuras de control
 - 2.2.7. Fuentes de datos

- 2.3. De los datos a la información
 - 2.3.1. Análisis de datos
 - 2.3.2. Tipos de análisis
 - 2.3.3. Extracción de información de un *dataset*
 - 2.3.3.1. Principales resúmenes estadísticos
 - 2.3.3.2. Análisis univariable
 - 2.3.3.3. Análisis multivariable
- 2.4. Visualización de datos y explicación de modelos de IA
 - 2.4.1. La visualización como herramienta de análisis
 - 2.4.2. Métodos de visualización
 - 2.4.3. Visualización de un conjunto de datos
 - 2.4.3.1. Análisis univariable
 - 2.4.3.2. Análisis multivariable
- 2.5. Calidad de los datos y detección de sesgos en IA
 - 2.5.1. Datos de calidad
 - 2.5.2. Limpieza de datos
 - 2.5.2.1. Análisis de valores duplicados
 - 2.5.2.2. Análisis de valores perdidos
 - 2.5.2.3. Análisis de valores extremos
 - 2.5.3. Preprocesamiento básico de datos
 - 2.5.3.1. Influencia y relación entre variables
 - 2.5.3.2. Conversión de variables categóricas
- 2.6. *Dataset* y generación sintética de datos
 - 2.6.1. Enriquecimiento del *dataset*
 - 2.6.2. La maldición de la dimensionalidad
 - 2.6.3. Modificación de nuestro conjunto de datos
- 2.7. Desbalanceo de clases y técnicas de reequilibrado
 - 2.7.1. Desbalanceo de clases
 - 2.7.2. Técnicas de mitigación del desbalanceo
 - 2.7.3. Balanceo de un *dataset*
 - 2.7.3.1. Ajuste de pesos
 - 2.7.3.2. Submuestreo
 - 2.7.3.3. Remuestreo

- 2.8. Modelos no supervisados
 - 2.8.1. Modelos no supervisados
 - 2.8.2. Métodos
 - 2.8.3. Clasificación con modelos no supervisados
- 2.9. Modelos supervisados
 - 2.9.1. Modelos supervisados
 - 2.9.2. Métodos
 - 2.9.3. Clasificación con modelos supervisados
 - 2.9.3.1. Fases del modelado
 - 2.9.3.2. División del conjunto de datos
 - 2.9.3.3. Métricas
 - 2.9.3.4. Aplicación de técnicas
- 2.10. Herramientas y buenas prácticas
 - 2.10.1. Buenas prácticas para un científico de datos
 - 2.10.2. El mejor modelo
 - 2.10.3. Herramientas útiles
- 3.4. Los valores perdidos y su tratamiento en el entrenamiento de modelos
 - 3.4.1. Tratamiento de valores perdidos
 - 3.4.2. Métodos de imputación de máxima verosimilitud
 - 3.4.3. Imputación de valores perdidos usando aprendizaje automático
- 3.5. El ruido en los datos y su influencia en la precisión de algoritmos
 - 3.5.1. Clases de ruido y atributos
 - 3.5.2. Filtrado de ruido
 - 3.5.3. El efecto del ruido
- 3.6. La maldición de la dimensionalidad y técnicas de reducción para IA
 - 3.6.1. *Oversampling*
 - 3.6.2. *Undersampling*
 - 3.6.3. Reducción de datos multidimensionales
- 3.7. De atributos continuos a discretos
 - 3.7.1. Datos continuos versus discretos
 - 3.7.2. Proceso de discretización
- 3.8. Los datos
 - 3.8.1. Selección de datos
 - 3.8.2. Perspectivas y criterios de selección
 - 3.8.3. Métodos de selección
- 3.9. Selección de instancias
 - 3.9.1. Métodos para la selección de instancias
 - 3.9.2. Selección de prototipos
 - 3.9.3. Métodos avanzados para la selección de instancias
- 3.10. Procesamiento de datos en entornos *Big Data*
 - 3.10.1. *Big Data*
 - 3.10.2. Preprocesamiento "clásico" Vs. Masivo
 - 3.10.3. Smart Data

Módulo 3. Minería de Datos. Selección, Preprocesamiento y Transformación

- 3.1. La inferencia estadística
 - 3.1.1. Estadística descriptiva Vs. Inferencia estadística
 - 3.1.2. Procedimientos paramétricos
 - 3.1.3. Procedimientos no paramétricos
- 3.2. Análisis exploratorio para identificar sesgos y patrones en datos de IA
 - 3.2.1. Análisis descriptivo
 - 3.2.2. Visualización
 - 3.2.3. Preparación de datos
- 3.3. Preparación de datos
 - 3.3.1. Integración de datos y limpieza de datos
 - 3.3.2. Normalización de datos
 - 3.3.3. Transformando atributos

Módulo 4. Sistemas Inteligentes

- 4.1. Teoría de agentes
 - 4.1.1. Agentes
 - 4.1.2. Agentes en Inteligencia Artificial
 - 4.1.3. Agentes en Ingeniería de Software
- 4.2. Arquitecturas de agentes
 - 4.2.1. El proceso de razonamiento de un agente
 - 4.2.2. Agentes reactivos
 - 4.2.3. Agentes deliberativos
 - 4.2.4. Agentes híbridos
 - 4.2.5. Comparativa
- 4.3. Representación de Información y conocimiento
 - 4.3.1. Distinción entre datos, información y conocimiento
 - 4.3.2. Evaluación de la calidad de los datos
 - 4.3.3. Métodos de captura de datos
 - 4.3.4. Métodos de adquisición de información
 - 4.3.5. Métodos de adquisición de conocimiento
- 4.4. IA explicable (XAI): principios, herramientas y aplicación en sistemas inteligentes
 - 4.4.1. Introducción a los metadatos
 - 4.4.2. Concepto filosófico de ontología
 - 4.4.3. Concepto informático de ontología
 - 4.4.4. Ontologías de dominio y ontologías de nivel superior
 - 4.4.5. Cómo construir una ontología
- 4.5. Ontologías
 - 4.5.1. Introducción a los metadatos
 - 4.5.2. Concepto filosófico de ontología
 - 4.5.3. Concepto informático de ontología
 - 4.5.4. Ontologías de dominio y ontologías de nivel superior
 - 4.5.5. Cómo construir una ontología
- 4.6. Lenguajes para ontologías y software para la creación de ontologías
 - 4.6.1. Tripletas RDF, Turtle y N3
 - 4.6.2. RDF Schema
 - 4.6.3. OWL
 - 4.6.4. SPARQL
 - 4.6.5. Introducción a las diferentes herramientas para la creación de ontologías
 - 4.6.6. Instalación y uso de Protégé
- 4.7. La web semántica
 - 4.7.1. El estado actual y futuro de la web semántica
 - 4.7.2. Aplicaciones de la web semántica
- 4.8. Otros modelos de representación del conocimiento
 - 4.8.1. Vocabularios
 - 4.8.2. Visión global
 - 4.8.3. Taxonomías
 - 4.8.4. Tesauros
 - 4.8.5. Folksonomías
 - 4.8.6. Estructura de Comparación
 - 4.8.7. Mapas mentales
- 4.9. Evaluación e integración de representaciones del conocimiento
 - 4.9.1. Lógica de orden cero
 - 4.9.2. Lógica de primer orden
 - 4.9.3. Lógica descriptiva
 - 4.9.4. Relación entre diferentes tipos de lógica
 - 4.9.5. Prolog: programación basada en lógica de primer orden
- 4.10. Razonadores semánticos, sistemas basados en conocimiento y Sistemas Expertos
 - 4.10.1. Concepto de razonador
 - 4.10.2. Aplicaciones de un razonador
 - 4.10.3. Sistemas basados en el conocimiento
 - 4.10.4. MYCIN, historia de los Sistemas Expertos
 - 4.10.5. Elementos y Arquitectura de Sistemas Expertos
 - 4.10.6. Creación de Sistemas Expertos



Módulo 5. Aprendizaje Automático

- 5.1. El conocimiento en bases de datos y su extracción mediante técnicas de IA
 - 5.1.1. Preprocesamiento de datos
 - 5.1.2. Análisis
 - 5.1.3. Interpretación y evaluación de los resultados
- 5.2. *Machine Learning*
 - 5.2.1. Aprendizaje supervisado y no supervisado
 - 5.2.2. Aprendizaje por refuerzo
 - 5.2.3. Aprendizaje semi - supervisado y otros modelos de aprendizaje
- 5.3. Clasificación
 - 5.3.1. Árboles de decisión y Aprendizaje basado en reglas
 - 5.3.2. Máquinas de Soporte Vectorial (SVM) y algoritmos de vecinos más cercanos (KNN)
 - 5.3.3. Métricas para algoritmos de clasificación
- 5.4. Regresión
 - 5.4.1. Regresión Lineal y regresión Logística
 - 5.4.2. Modelos de regresión no lineales
 - 5.4.3. Análisis de series temporales
 - 5.4.4. Métricas para algoritmos de regresión
- 5.5. *Clustering*
 - 5.5.1. Agrupamiento Jerárquico
 - 5.5.2. Agrupamiento Particional
 - 5.5.3. Métricas para algoritmos de *clustering*
- 5.6. Reglas de asociación
 - 5.6.1. Definición y medidas de interés
 - 5.6.2. Métodos de extracción de reglas
 - 5.6.3. Métricas para los algoritmos de reglas de asociación
- 5.7. Multiclasificadores y estrategias de combinación de modelos
 - 5.7.1. "*Bootstrap aggregation*" o "*bagging*"
 - 5.7.2. Algoritmo de "*random forests*"
 - 5.7.3. Algoritmo de "*boosting*"

- 5.8. Modelos de razonamiento probabilístico
 - 5.8.1. Razonamiento probabilístico
 - 5.8.2. Redes bayesianas o redes de creencia
 - 5.8.3. "Hidden Markov Models"
- 5.9. Perceptrón Multicapa
 - 5.9.1. Red neuronal
 - 5.9.2. Aprendizaje automático con redes neuronales
 - 5.9.3. Descenso del gradiente, "backpropagation" y funciones de activación
 - 5.9.4. Implementación de una red neuronal artificial
- 5.10. Aprendizaje profundo y arquitecturas neuronales avanzadas
 - 5.10.1. Redes neuronales profundas. Introducción
 - 5.10.2. Redes convolucionales
 - 5.10.3. Sequence Modeling
 - 5.10.4. Tensorflow y Pytorch

Módulo 6. Las Redes Neuronales, Base de Deep Learn

- 6.1. Aprendizaje Profundo
 - 6.1.1. Tipos de aprendizaje profundo
 - 6.1.2. Aplicaciones del aprendizaje profundo
 - 6.1.3. Ventajas y desventajas del aprendizaje profundo
- 6.2. Operaciones
 - 6.2.1. Suma
 - 6.2.2. Resta
 - 6.2.3. Traslado
- 6.3. Capas
 - 6.3.1. Capa de entrada
 - 6.3.2. Capa oculta
 - 6.3.3. Capa de salida
- 6.4. Unión de Capas y Operaciones
 - 6.4.1. Diseño de arquitecturas
 - 6.4.2. Conexión entre capas
 - 6.4.3. Propagación hacia adelante

- 6.5. Construcción de la primera red neuronal
 - 6.5.1. Diseño de la red
 - 6.5.2. Establecer los pesos
 - 6.5.3. Entrenamiento de la red
- 6.6. Entrenador, optimizador y función de pérdida en el proceso de aprendizaje
 - 6.6.1. Selección del optimizador
 - 6.6.2. Establecimiento de una función de pérdida
 - 6.6.3. Establecimiento de una métrica
- 6.7. Aplicación de los Principios de las Redes Neuronales
 - 6.7.1. Funciones de activación
 - 6.7.2. Propagación hacia atrás
 - 6.7.3. Ajuste de los parámetros
- 6.8. De las neuronas biológicas a las artificiales
 - 6.8.1. Funcionamiento de una neurona biológica
 - 6.8.2. Transferencia de conocimiento a las neuronas artificiales
 - 6.8.3. Establecer relaciones entre ambas
- 6.9. Implementación de MLP (Perceptrón multicapa) con Keras y evaluación del modelo
 - 6.9.1. Definición de la estructura de la red
 - 6.9.2. Compilación del modelo
 - 6.9.3. Entrenamiento del modelo
- 6.10. Hiperparámetros de Fine tuning de Redes Neuronales
 - 6.10.1. Selección de la función de activación
 - 6.10.2. Establecer el learning rate
 - 6.10.3. Ajuste de los pesos

Módulo 7. Entrenamiento de Redes Neuronales Profundas

- 7.1. Problemas de gradientes y soluciones en redes profundas
 - 7.1.1. Técnicas de optimización de gradiente
 - 7.1.2. Gradientes Estocásticos
 - 7.1.3. Técnicas de inicialización de pesos
- 7.2. Reutilización de capas preentrenadas
 - 7.2.1. Entrenamiento de transferencia de aprendizaje
 - 7.2.2. Extracción de características
 - 7.2.3. Aprendizaje profundo

- 7.3. Optimizadores y su impacto en la convergencia del modelo
 - 7.3.1. Optimizadores de descenso de gradiente estocástico
 - 7.3.2. Optimizadores Adam y RMSprop
 - 7.3.3. Optimizadores de momento
- 7.4. Programación de la tasa de aprendizaje
 - 7.4.1. Control de tasa de aprendizaje automático
 - 7.4.2. Ciclos de aprendizaje
 - 7.4.3. Términos de suavizado
- 7.5. Sobreajuste
 - 7.5.1. Validación cruzada
 - 7.5.2. Regularización
 - 7.5.3. Métricas de evaluación
- 7.6. Directrices prácticas
 - 7.6.1. Diseño de modelos
 - 7.6.2. Selección de métricas y parámetros de evaluación
 - 7.6.3. Pruebas de hipótesis
- 7.7. *Transfer learning*
 - 7.7.1. Entrenamiento de transferencia de aprendizaje
 - 7.7.2. Extracción de características
 - 7.7.3. Aprendizaje profundo
- 7.8. *Data Augmentation*
 - 7.8.1. Transformaciones de imagen
 - 7.8.2. Generación de datos sintéticos
 - 7.8.3. Transformación de texto
- 7.9. Aplicación práctica de *Transfer Learning*
 - 7.9.1. Entrenamiento de transferencia de aprendizaje
 - 7.9.2. Extracción de características
 - 7.9.3. Aprendizaje profundo
- 7.10. Regularización
 - 7.10.1. L1 y L2
 - 7.10.2. Regularización por máxima entropía
 - 7.10.3. *Dropout*

Módulo 8. Personalización de Modelos y Entrenamientos con TensorFlow

- 8.1. TensorFlow
 - 8.1.1. Uso de la biblioteca TensorFlow
 - 8.1.2. Entrenamiento de modelos con TensorFlow
 - 8.1.3. Operaciones con gráficos en TensorFlow
- 8.2. TensorFlow y NumPy
 - 8.2.1. Entorno computacional NumPy para TensorFlow
 - 8.2.2. Utilización de los arrays NumPy con TensorFlow
 - 8.2.3. Operaciones NumPy para los gráficos de TensorFlow
- 8.3. Personalización de modelos y algoritmos de entrenamiento
 - 8.3.1. Construcción de modelos personalizados con TensorFlow
 - 8.3.2. Gestión de parámetros de entrenamiento
 - 8.3.3. Utilización de técnicas de optimización para el entrenamiento
- 8.4. Funciones y gráficos de TensorFlow
 - 8.4.1. Funciones con TensorFlow
 - 8.4.2. Utilización de gráficos para el entrenamiento de modelos
 - 8.4.3. Optimización de gráficos con operaciones de TensorFlow
- 8.5. Carga y preprocesamiento de datos con TensorFlow
 - 8.5.1. Carga de conjuntos de datos con TensorFlow
 - 8.5.2. Preprocesamiento de datos con TensorFlow
 - 8.5.3. Utilización de herramientas de TensorFlow para la manipulación de datos
- 8.6. La API tf.data y el formato TFRecord
 - 8.6.1. Utilización de la API tf.data para el procesamiento de datos
 - 8.6.2. Construcción de flujos de datos con tf.data
 - 8.6.3. Uso de la API tf.data para el entrenamiento de modelos
- 8.7. Uso de TensorFlow en modelos generativos: GANs y Autoencoders
 - 8.7.1. Utilización de la API TFRecord para la serialización de datos
 - 8.7.2. Carga de archivos TFRecord con TensorFlow
 - 8.7.3. Utilización de archivos TFRecord para el entrenamiento de modelos
- 8.8. Capas de preprocesamiento de Keras
 - 8.8.1. Utilización de la API de preprocesamiento de Keras
 - 8.8.2. Construcción de pipeline de preprocesamiento con Keras
 - 8.8.3. Uso de la API de preprocesamiento de Keras para el entrenamiento de modelos

- 8.9. El proyecto TensorFlow Datasets
 - 8.9.1. Utilización de TensorFlow *Datasets* para la carga de datos
 - 8.9.2. Preprocesamiento de datos con TensorFlow *Datasets*
 - 8.9.3. Uso de TensorFlow *Datasets* para el entrenamiento de modelos
- 8.10. Construcción de una Aplicación de *Deep Learning* con TensorFlow. Aplicación Práctica
 - 8.10.1. Construcción de una aplicación de *Deep Learning* con TensorFlow
 - 8.10.2. Entrenamiento de un modelo con TensorFlow
 - 8.10.3. Utilización de la aplicación para la predicción de resultados

Módulo 9. Procesamiento del Lenguaje Natural (NLP)

- 9.1. Generación de texto utilizando RNN
 - 9.1.1. Entrenamiento de una RNN para generación de texto
 - 9.1.2. Generación de lenguaje natural con RNN
 - 9.1.3. Aplicaciones de generación de texto con RNN
- 9.2. Creación del conjunto de datos de entrenamiento
 - 9.2.1. Preparación de los datos para el entrenamiento de una RNN
 - 9.2.2. Almacenamiento del conjunto de datos de entrenamiento
 - 9.2.3. Limpieza y transformación de los datos
- 9.3. Clasificación de opiniones con RNN
 - 9.3.1. Clasificación de opiniones con RNN
 - 9.3.2. Detección de temas en los comentarios
 - 9.3.3. Análisis de sentimiento con algoritmos de aprendizaje profundo
- 9.4. Red codificador - decodificador para traducción automática neuronal
 - 9.4.1. Entrenamiento de una RNN para la traducción automática
 - 9.4.2. Uso de una red *encoder - decoder* para la traducción automática
 - 9.4.3. Mejora de la precisión de la traducción automática con RNN
- 9.5. Mecanismos de atención
 - 9.5.1. Aplicación de mecanismos de atención en RNN
 - 9.5.2. Uso de mecanismos de atención para mejorar la precisión de los modelos
 - 9.5.3. Ventajas de los mecanismos de atención en las redes neuronales



- 9.6. Modelos Transformers
 - 9.6.1. Uso de los modelos Transformers para procesamiento de lenguaje natural
 - 9.6.2. Aplicación de los modelos Transformers para visión
 - 9.6.3. Ventajas de los modelos Transformers
- 9.7. Transformers para visión
 - 9.7.1. Uso de los modelos Transformers para visión
 - 9.7.2. Preprocesamiento de los datos de imagen
 - 9.7.3. Entrenamiento de un modelo Transformer para visión
- 9.8. Librería de Transformers de Hugging Face
 - 9.8.1. Uso de la librería de Transformers de Hugging Face
 - 9.8.2. Aplicación de la librería de Transformers de Hugging Face
 - 9.8.3. Ventajas de la librería de Transformers de Hugging Face
- 9.1. Comparativa entre librerías de Transformers: criterios de selección y rendimiento
 - 9.9.1. Comparación entre las distintas librerías de Transformers
 - 9.9.2. Uso de las demás librerías de
 - 9.9.3. Ventajas de las demás librerías de Transformers
- 9.10. Desarrollo de una aplicación de NLP combinando RNN, atención y Transformers
 - 9.10.1. Desarrollo de una aplicación de procesamiento de lenguaje natural con RNN y atención
 - 9.10.2. Uso de RNN, mecanismos de atención y modelos Transformers en la aplicación
 - 9.10.3. Evaluación de la aplicación práctica

Módulo 10. Computación Bioinspirada

- 10.1. Introducción a la computación bioinspirada
 - 10.1.1. Definición de los algoritmos bioinspirados. Características principales
 - 10.1.2. Metodología de algoritmos bioinspirados
 - 10.1.3. Aplicación
- 10.2. Algoritmos de adaptación social
 - 10.2.1. Computación bioinspirada basada en colonia de hormigas
 - 10.2.2. Variantes de los algoritmos de colonias de hormigas
 - 10.2.3. Computación basada en nubes de partículas

- 10.3. Algoritmos genéticos
 - 10.3.1. Estructura general
 - 10.3.2. Implementaciones de los principales operadores
- 10.4. Estrategias de exploración-explotación del espacio para algoritmos genético
 - 10.4.1. Algoritmo CHC
 - 10.4.2. Problemas multimodales
- 10.5. Modelos de computación evolutiva (I)
 - 10.5.1. Estrategias evolutivas
 - 10.5.2. Programación evolutiva
 - 10.5.3. Algoritmos basados en evolución diferencial
- 10.6. Modelos de computación evolutiva (II)
 - 10.6.1. Modelos de evolución basados en estimación de distribuciones (EDA)
 - 10.6.2. Programación genética
- 10.7. Programación evolutiva aplicada a problemas de aprendizaje
 - 10.7.1. Aprendizaje basado en reglas
 - 10.7.2. Métodos evolutivos en problemas de selección de instancias
- 10.8. Problemas multiobjetivo
 - 10.8.1. Concepto de dominancia
 - 10.8.2. Aplicación de algoritmos evolutivos a problemas multiobjetivo
- 10.9. Optimización evolutiva de redes neuronales
 - 10.9.1. Optimización evolutiva de redes neuronales
 - 10.9.2. Neuroevolución de Topologías: algoritmos y metodologías avanzadas
 - 10.9.3. Aplicaciones prácticas y casos de éxito
 - 10.9.4. Desafíos, direcciones futuras y su intersección con el *Deep Learning*
- 10.10. Modelos híbridos neuro - evolutivos
 - 10.10.1. Fundamentos y motivación de la hibridación
 - 10.10.2. Estrategias de hibridación neuro-evolutiva
 - 10.10.3. Arquitecturas y *frameworks* comunes
 - 10.10.4. Integración de algoritmos evolutivos y redes neuronales
 - 10.10.5. Implementación y consideraciones prácticas
 - 10.10.6. Aplicaciones y casos de uso avanzados
 - 10.10.7. Desafíos y direcciones futuras

Módulo 11. Mejora de la Productividad en Desarrollo de Software con Inteligencia Artificial

- 11.1. Preparar un entorno de desarrollo adecuado
 - 11.1.1. Preparar un entorno de desarrollo adecuado
 - 11.1.2. Configuración de las herramientas elegidas
 - 11.1.3. Implementación de pipelines de CI/CD adaptados a proyectos con IA
 - 11.1.4. Gestión eficiente de dependencias y versiones en entornos de desarrollo
- 11.2. Extensiones imprescindibles de IA para Visual Studio Code
 - 11.2.1. Exploración y selección de extensiones de IA para Visual Studio Code
 - 11.2.2. Integración de herramientas de análisis estático y dinámico en el IDE
 - 11.2.3. Automatización de tareas repetitivas con extensiones específicas
 - 11.2.4. Personalización del entorno de desarrollo para mejorar la eficiencia
- 11.3. Diseño No - code de Interfaces de Usuario con elementos de IA
 - 11.3.1. Principios del diseño no - code y su aplicación en interfaces de usuario
 - 11.3.2. Incorporación de elementos de IA en el diseño visual de interfaces
 - 11.3.3. Herramientas y plataformas para la creación no - code de interfaces inteligentes
 - 11.3.4. Evaluación y mejora continua de interfaces no - code con IA
- 11.4. Optimización de código usando ChatGPT
 - 11.4.1. Identificar código duplicado
 - 11.4.2. Refactorizar
 - 11.4.3. Crear códigos legibles
 - 11.4.4. Entender lo que hace un código
 - 11.4.5. Mejorar nombre de variables y funciones
 - 11.4.6. Creación de documentación automática
- 11.5. Gestión de repositorios con IA
 - 11.5.1. Automatización de procesos de control de versiones con técnicas de IA
 - 11.5.2. Detección de conflictos y resolución automática en entornos colaborativos
 - 11.5.3. Análisis predictivo de cambios y tendencias en repositorios de código
 - 11.5.4. Mejoras en la organización y categorización de repositorios mediante IA
- 11.6. Integración de IA en gestión con bases de datos con ChatGPT
 - 11.6.1. Optimización de consultas y rendimiento utilizando técnicas de IA
 - 11.6.2. Análisis predictivo de patrones de acceso a bases de datos
 - 11.6.3. Implementación de sistemas de recomendación para optimizar la estructura de la base de datos
 - 11.6.4. Monitoreo y detección proactiva de posibles problemas en bases de datos
- 11.7. Búsqueda de fallos y creación de test unitarios con IA
 - 11.7.1. Generación automática de casos de prueba mediante técnicas de IA
 - 11.7.2. Detección temprana de vulnerabilidades y errores utilizando análisis estático con IA
 - 11.7.3. Mejora de la cobertura de pruebas mediante la identificación de áreas críticas por IA
- 11.8. *Pair Programming* con GitHub Copilot
 - 11.8.1. Integración y uso efectivo de GitHub Copilot en sesiones de *Pair Programming*
 - 11.8.2. Mejoras en la comunicación y colaboración entre desarrolladores con GitHub Copilot
 - 11.8.3. Estrategias para aprovechar al máximo las sugerencias de código generadas por GitHub Copilot
 - 11.8.4. Casos de estudio y buenas prácticas en *Pair Programming* asistido por IA
- 11.9. Traducción automática entre lenguajes de programación
 - 11.9.1. Herramientas y servicios de traducción automática específicos para lenguajes de programación
 - 11.9.2. Adaptación de algoritmos de traducción automática a contextos de desarrollo
 - 11.9.3. Mejora de la interoperabilidad entre diferentes lenguajes mediante traducción automática
 - 11.9.4. Evaluación y mitigación de posibles desafíos y limitaciones en la traducción automática
- 11.10. Herramientas de IA recomendadas para mejorar la productividad
 - 11.10.1. Análisis comparativo de herramientas de IA para el desarrollo de software
 - 11.10.2. Integración de herramientas de IA en flujos de trabajo
 - 11.10.3. Automatización de tareas rutinarias con herramientas de IA
 - 11.10.4. Evaluación y selección de herramientas basada en el contexto y los requerimientos del proyecto

Módulo 12. Arquitectura del software con Inteligencia Artificial

- 12.1. Optimización y gestión del rendimiento en herramientas con IA
 - 12.1.1. Análisis y perfilado de rendimiento en herramientas con IA
 - 12.1.2. Estrategias de optimización de algoritmos y modelos de IA
 - 12.1.3. Implementación de técnicas de *caching* y paralelización para mejorar el rendimiento
 - 12.1.4. Herramientas y metodologías para la monitorización continua del rendimiento en tiempo real
- 12.2. Escalabilidad en aplicaciones de IA
 - 12.2.1. Diseño de arquitecturas escalables para aplicaciones de IA
 - 12.2.2. Implementación de técnicas de particionamiento y distribución de carga
 - 12.2.3. Manejo de flujos de trabajo y carga de trabajo en sistemas escalables
 - 12.2.4. Estrategias para la expansión horizontal y vertical en entornos con demanda variable
- 12.3. Mantenibilidad de aplicaciones con IA
 - 12.3.1. Principios de diseño para facilitar la mantenibilidad en proyectos de IA
 - 12.3.2. Estrategias de documentación específicas para modelos y algoritmos de IA
 - 12.3.3. Implementación de pruebas unitarias y de integración para facilitar el mantenimiento
 - 12.3.4. Métodos para la refactorización y mejora continua en sistemas con componentes de IA
- 12.4. Diseño de sistemas de gran escala
 - 12.4.1. Principios arquitectónicos para el diseño de sistemas de gran escala
 - 12.4.2. Descomposición de sistemas complejos en microservicios
 - 12.4.3. Implementación de patrones de diseño específicos para sistemas distribuidos
 - 12.4.4. Estrategias para la gestión de la complejidad en arquitecturas de gran escala con componentes de IA
- 12.5. Almacenamiento de datos de gran escala para herramientas de IA
 - 12.5.1. Selección de tecnologías de almacenamiento de datos escalables
 - 12.5.2. Diseño de esquemas de bases de datos para el manejo eficiente de grandes volúmenes de datos
 - 12.5.3. Estrategias de particionamiento y replicación en entornos de almacenamiento de datos masivos
 - 12.5.4. Implementación de sistemas de gestión de datos para garantizar la integridad y disponibilidad en proyectos con IA
- 12.6. Estructuras de datos Con IA
 - 12.6.1. Adaptación de estructuras de datos clásicas para su uso en algoritmos de IA
 - 12.6.2. Diseño y optimización de estructuras de datos específicas con ChatGPT
 - 12.6.3. Integración de estructuras de datos eficientes en sistemas con procesamiento intensivo de datos
 - 12.6.4. Estrategias para la manipulación y almacenamiento de datos en tiempo real en estructuras de datos con IA
- 12.7. Algoritmos de programación para productos con IA
 - 12.7.1. Desarrollo e implementación de algoritmos específicos para aplicaciones con IA
 - 12.7.2. Estrategias de selección de algoritmos, según el tipo de problema y los requisitos del producto
 - 12.7.3. Adaptación de algoritmos clásicos para su integración en sistemas de Inteligencia Artificial
 - 12.7.4. Evaluación y comparación de rendimiento entre diferentes algoritmos en contextos de desarrollo con IA
- 12.8. Patrones diseño para desarrollo con IA
 - 12.8.1. Identificación y aplicación de patrones de diseño comunes en proyectos con componentes de IA
 - 12.8.2. Desarrollo de patrones específicos para la integración de modelos y algoritmos en sistemas existentes
 - 12.8.3. Estrategias de implementación de patrones para mejorar la reusabilidad y mantenibilidad en proyectos de IA
 - 12.8.4. Casos de estudio y buenas prácticas en la aplicación de patrones de diseño en arquitecturas con IA
- 12.9. Implementación de *clean architecture*
 - 12.9.1. Principios y conceptos fundamentales de *clean architecture*
 - 12.9.2. Adaptación de *clean architecture* a proyectos con componentes de IA
 - 12.9.3. Implementación de capas y dependencias en sistemas con *clean architecture*
 - 12.9.4. Beneficios y desafíos de la implementación de *clean architecture* en el desarrollo de software con IA

- 12.10. Desarrollo de software seguro en aplicaciones web con IA
 - 12.10.1. Principios de seguridad en el desarrollo de software con componentes de IA
 - 12.10.2. Identificación y mitigación de posibles vulnerabilidades en modelos y algoritmos de IA
 - 12.10.3. Implementación de prácticas de desarrollo seguro en aplicaciones web con funcionalidades de Inteligencia Artificial
 - 12.10.4. Estrategias para la protección de datos sensibles y la prevención de ataques en proyectos con IA

Módulo 13. Proyectos web con Inteligencia Artificial

- 13.1. Preparación del entorno de trabajo para desarrollo web con IA
 - 13.1.1. Configuración de entornos de desarrollo web para proyectos con Inteligencia Artificial
 - 13.1.1.1. Editores de código
 - 13.1.1.2. Gestión de versiones
 - 13.1.1.3. Configuración de entornos virtuales
 - 13.1.1.4. Automatización de tareas
 - 13.1.2. Selección y preparación de herramientas esenciales para el desarrollo web con IA
 - 13.1.2.1. Lenguajes de programación
 - 13.1.2.2. Bases de datos
 - 13.1.3. Integración de bibliotecas y *frameworks* específicos para proyectos web con Inteligencia Artificial
 - 13.1.3.1. *Frameworks*
 - 13.1.4. Implementación de buenas prácticas en la configuración de entornos de desarrollo colaborativos
 - 13.1.4.1. Buenas prácticas
 - 13.1.4.2. Implementación
- 13.2. Creación *workspace* para proyectos de IA
 - 13.2.1. Diseño y organización efectiva de *workspaces* para proyectos web con componentes de Inteligencia Artificial
 - 13.2.1.1. Estructura del *Workspace*
 - 13.2.1.2. Herramientas de IA
 - 13.2.1.3. GitHub copilot
 - 13.2.1.4. Buenas prácticas
 - 13.2.2. Uso de herramientas de gestión de proyectos y control de versiones en el *workspace*
 - 13.2.2.1. Herramientas
 - 13.2.2.2. Control de versiones
 - 13.2.2.3. Flujo de trabajo típico con Git y GitHub
 - 13.2.2.4. Automatización de tareas
 - 13.2.2.5. Documentación y trazabilidad
 - 13.2.3. Estrategias para la colaboración y comunicación eficientes en el equipo de desarrollo
 - 13.2.3.1. Roles y responsabilidades
 - 13.2.3.2. Herramientas de comunicación
 - 13.2.3.3. Buenas prácticas para la comunicación
 - 13.2.3.4. Metodologías ágiles Implementación en proyectos de IA Reuniones y rituales
 - 13.2.4. Adaptación del *workspace* a las necesidades específicas de proyectos web con IA
 - 13.2.4.1. Personalización del *Workspace*
 - 13.2.4.2. Escalabilidad y flexibilidad
 - 13.2.4.3. Seguridad y cumplimiento
 - 13.2.4.4. Evaluación y mejora Continua
- 13.3. Patrones de diseño en productos con IA
 - 13.3.1. Identificación y aplicación de patrones de diseño comunes en interfaces de usuario con elementos de Inteligencia Artificial
 - 13.3.2. Patrones de Diseño Comunes
 - 13.3.2.1. Modelo-Vista-Controlador (MVC)
 - 13.3.2.2. Modelo-Vista-VistaModelo (MVVM)
 - 13.3.2.3. VIPER
 - 13.3.2.4. Clean Architecture
 - 13.3.2.5. Microservicios
 - 13.3.2.6. Arquitectura Hexagonal (Ports and Adapters)
 - 13.3.3. Desarrollo de patrones específicos para mejorar la experiencia de usuario en proyectos web con IA
 - 13.3.3.1. Conceptos de UX
 - 13.3.3.2. Importancia de UX en Proyectos con IA



- 13.3.4. Integración de patrones de diseño en la arquitectura general de proyectos web con Inteligencia Artificial
 - 13.3.4.1. Arquitectura de Software y Patrones de Diseño
 - 13.3.4.2. Integración de Patrones de Diseño Clásicos
 - 13.3.4.3. Incorporación de Patrones de IA
 - 13.3.4.4. Uso de Microservicios
 - 13.3.4.5. Integración Continua y Despliegue (CI/CD)
 - 13.3.4.6. Ejemplos de Integración Exitosa
- 13.3.5. Evaluación y selección de patrones de diseño adecuados según el contexto del proyecto
 - 13.3.5.1. Criterios para la Evaluación de Patrones de Diseño
 - 13.3.5.2. Proceso de Selección de Patrones de Diseño
 - 13.3.5.3. Caso de Estudio: Selección de Patrones para una Aplicación de E-commerce
- 13.4. Desarrollo *Frontend* con IA
 - 13.4.1. Integración de modelos de IA en la capa de presentación de proyectos web
 - 13.4.1.1. Ejemplos prácticos de integración de modelos de IA en aplicaciones web
 - 13.4.1.2. Herramientas y *frameworks* compatibles con GitHub Copilot para la integración de IA
 - 13.4.2. Desarrollo de interfaces de usuario adaptativas con elementos de Inteligencia Artificial
 - 13.4.2.1. Principios del diseño adaptativo y su importancia en la experiencia de usuario
 - 13.4.2.2. Uso de GitHub Copilot para generar código adaptativo
 - 13.4.2.3. Ejemplos de interfaces adaptativas
 - 13.4.2.4. Técnicas de IA para la personalización
 - 13.4.3. Implementación de funcionalidades de procesamiento de lenguaje natural (PLN) en el *frontend*
 - 13.4.3.1. Introducción al procesamiento de lenguaje natural y sus aplicaciones en *frontend*
 - 13.4.3.2. Ejemplos de implementación de PLN utilizando GitHub Copilot
 - 13.4.3.3. Funcionalidades de PLN en *frontend*
 - 13.4.3.4. Integración de APIs de PLN en proyectos *frontend*

- 13.4.4. Estrategias para la optimización del rendimiento en el desarrollo *frontend* con IA
 - 13.4.4.1. Análisis de los factores que afectan el rendimiento en aplicaciones *frontend* con IA
 - 13.4.4.2. Herramientas y técnicas para el monitoreo y la optimización del rendimiento
 - 13.4.4.3. Ejemplos prácticos de optimización
 - 13.4.4.4. Buenas prácticas para mantener un equilibrio entre funcionalidad y rendimiento
- 13.5. Creación de base de dato
 - 13.5.1. Selección de tecnologías de bases de datos para proyectos web con inteligencia artificial
 - 13.5.1.1. Evaluación de diferentes tecnologías de bases de datos
 - 13.5.1.2. Consideraciones para la selección de la tecnología adecuada
 - 13.5.1.3. Comparación entre bases de datos relacionales y no relacionales en el contexto de proyectos de IA
 - 13.5.2. Diseño de esquemas de bases de datos para almacenar y gestionar datos relacionados con IA
 - 13.5.2.1. Conceptos básicos de diseño de bases de datos
 - 13.5.2.2. Modelado de datos para proyectos de IA
 - 13.5.2.3. Técnicas avanzadas de diseño de esquemas
 - 13.5.3. Implementación de sistemas de almacenamiento eficientes para grandes volúmenes de datos generados por modelos de IA
 - 13.5.3.1. Estrategias de almacenamiento para manejar grandes volúmenes de datos
 - 13.5.3.2. Uso de tecnologías de almacenamiento distribuido y escalable
 - 13.5.3.3. Optimización de consultas para mejorar el rendimiento en entornos de IA con grandes conjuntos de datos
 - 13.5.4. Estrategias para la seguridad y protección de datos sensibles en bases de datos de proyectos web con IA
 - 13.5.4.1. Principios de seguridad de bases de datos
 - 13.5.4.2. Identificación de riesgos de seguridad específicos en proyectos web con IA
 - 13.5.4.3. Implementación de medidas de seguridad
- 13.6. Desarrollo *Backend* con IA
 - 13.6.1. Integración de servicios y modelos de IA en la lógica de negocio del *backend*
 - 13.6.2. Desarrollo de APIs y *endpoints* específicos para la comunicación entre el *frontend* y los componentes de IA
 - 13.6.3. Implementación de lógica de procesamiento de datos y toma de decisiones en el backend con Inteligencia Artificial
 - 13.6.4. Estrategias para la escalabilidad y rendimiento en el desarrollo *backend* de proyectos web con IA
- 13.7. Optimizar el proceso de despliegue de tu web
 - 13.7.1. Automatización de procesos de construcción y despliegue de proyectos web con ChatGPT
 - 13.7.1.1. Importancia de la automatización en el proceso de desarrollo y despliegue
 - 13.7.1.2. Introducción a ChatGPT y su aplicación en la automatización de procesos
 - 13.7.2. Implementación de pipelines de CI/CD adaptados a aplicaciones web con GitHub Copilot
 - 13.7.2.1. Conceptos básicos de CI/CD y su importancia en el desarrollo de aplicaciones web
 - 13.7.2.2. Introducción a GitHub Copilot y su aplicación en la implementación de pipelines de CI/CD adaptados a aplicaciones web
 - 13.7.3. Estrategias para la gestión eficiente de versiones y actualizaciones en despliegues continuos
 - 13.7.3.1. Importancia de la gestión de versiones y actualizaciones en despliegues continuos
 - 13.7.3.2. Estrategias para gestionar eficazmente versiones y actualizaciones en entornos de despliegues continuos
 - 13.7.3.3. Herramientas y prácticas recomendadas para la gestión de versiones y actualizaciones en proyectos web
 - 13.7.4. Monitoreo y análisis post-despliegue para la mejora continua del proceso
 - 13.7.4.1. Concepto de monitoreo y análisis post-despliegue y su importancia
 - 13.7.4.2. Herramientas y técnicas para realizar monitoreo y análisis post-despliegue en proyectos web
 - 13.7.4.3. Estrategias para utilizar los resultados del monitoreo y análisis post-despliegue

- 13.8. IA en la computación en nube
 - 13.8.1. Integración de servicios de inteligencia artificial en plataformas de computación en la nube
 - 13.8.1.1. Conceptos fundamentales de la integración de IA en la nube
 - 13.8.1.2. Principales plataformas y servicios de nube con capacidades de IA
 - 13.8.1.3. Ejemplos de casos de uso y aplicaciones prácticas
 - 13.8.2. Desarrollo de soluciones escalables y distribuidas utilizando servicios de nube con capacidades de IA
 - 13.8.3. Estrategias para el manejo eficiente de recursos y costos en entornos de nube con aplicaciones web con IA
 - 13.8.4. Evaluación y comparación de proveedores de servicios en la nube para proyectos web con Inteligencia Artificial
- 13.9. Creación de un proyecto con IA para entornos LAMP
 - 13.9.1. Adaptación de proyectos web basados en la pila LAMP para incluir componentes de inteligencia artificial
 - 13.9.2. Integración de bibliotecas y frameworks específicos de IA en entornos LAMP
 - 13.9.3. Desarrollo de funcionalidades de IA que complementan la arquitectura LAMP tradicional
 - 13.9.4. Estrategias para la optimización y mantenimiento en proyectos web con IA en entornos LAMP
 - 13.9.4.1. Ejemplo Práctico
 - 13.9.4.2. Código PHP para conectarse a MySQL y mostrar datos en una página web
- 13.10. Creación de un proyecto con IA para entornos MEVN
 - 13.10.1. Integración de tecnologías y herramientas de la pila MEVN con componentes de inteligencia artificial
 - 13.10.1.1. MongoDB
 - 13.10.1.2. Express.js
 - 13.10.1.3. Vue.js
 - 13.10.1.4. Node.js

- 13.10.2. Desarrollo de aplicaciones web modernas y escalables en entornos MEVN con capacidades de IA
- 13.10.3. Implementación de funcionalidades de procesamiento de datos y aprendizaje automático en proyectos MEVN
- 13.10.4. Estrategias para la mejora del rendimiento y la seguridad en aplicaciones web con IA en entornos MEVN
- 13.10.5. Ejemplo práctico

Módulo 14. Aplicaciones móviles con Inteligencia Artificial

- 14.1. Preparación de entorno de trabajo para desarrollo móvil con IA
 - 14.1.1. Configuración de entornos de desarrollo móvil para proyectos con Inteligencia Artificial
 - 14.1.2. Selección y preparación de herramientas específicas para el desarrollo de aplicaciones móviles con IA
 - 14.1.3. Integración de bibliotecas y *frameworks* de IA en entornos de desarrollo móvil
 - 14.1.4. Configuración de emuladores y dispositivos reales para pruebas de aplicaciones móviles con componentes de Inteligencia Artificial
- 14.2. Creación de un *workspace* con Github copilot
 - 14.2.1. Integración de GitHub Copilot en entornos de desarrollo móvil
 - 14.2.1.1. Integración de GitHub Copilot en entornos de desarrollo móvil
 - 14.2.1.2. Instalación en Android Studio
 - 14.2.1.3. Configuración en Xcode
 - 14.2.1.4. Uso en Visual Studio Code
 - 14.2.1.5. Uso efectivo de GitHub Copilot para la generación de código en proyectos con IA
 - 14.2.1.6. Estrategias para la colaboración entre desarrolladores al utilizar GitHub Copilot en el *workspace*
 - 14.2.1.7. Buenas prácticas y limitaciones en el uso de GitHub Copilot en el desarrollo de aplicaciones móviles con IA
 - 14.2.2. Uso efectivo de GitHub Copilot para la generación de código en proyectos con IA
 - 14.2.2.1. Optimización del código generado

- 14.2.3. Estrategias para la colaboración entre desarrolladores al utilizar GitHub Copilot en el *workspace*
 - 14.2.3.1. Creación de un repositorio compartido
 - 14.2.3.2. Permisos y accesos Integración de Copilot
 - 14.2.3.3. Estrategias de resolución de conflictos Revisión de código
 - 14.2.3.4. Herramientas de fusión
 - 14.2.3.5. Asignación de tareas Colaboración efectiva
- 14.2.4. Buenas prácticas y limitaciones en el uso de GitHub Copilot en el desarrollo de aplicaciones móviles con IA
 - 14.2.4.1. Buenas prácticas
 - 14.2.4.2. Limitaciones
 - 14.2.4.3. Consideraciones éticas y de seguridad
- 14.3. Configuración de Firebase
 - 14.3.1. Configuración inicial de un proyecto en Firebase para el desarrollo móvil
 - 14.3.1.1. Configuración inicial de Firebase
 - 14.3.2. Integración de Firebase en aplicaciones móviles con funcionalidades de Inteligencia Artificial
 - 14.3.3. Uso de servicios de Firebase como base de datos, autenticación y notificaciones en proyectos con IA
 - 14.3.4. Estrategias para la gestión de datos y eventos en tiempo real en aplicaciones móviles con Firebase
- 14.4. Conceptos de *clean architecture*, *DataSources*, repositories
 - 14.4.1. Principios fundamentales de *clean architecture* en el desarrollo móvil con IA
 - 14.4.2. Implementación de capas de *DataSources* y repositories con GitHub Copilot
 - 14.4.3. Diseño y estructuración de componentes en proyectos móviles con GitHub Copilot
 - 14.4.4. Beneficios y desafíos de la implementación de *clean architecture* en aplicaciones móviles con IA



- 14.5. Creación de pantalla de autenticación
 - 14.5.1. Diseño y desarrollo de interfaces de usuario para pantallas de autenticación en aplicaciones móviles con IA
 - 14.5.2. Integración de servicios de autenticación con *firebase* en la pantalla de inicio de sesión
 - 14.5.3. Uso de técnicas de seguridad y protección de datos en la pantalla de autenticación
 - 14.5.4. Personalización y adaptación de la experiencia de usuario en la pantalla de autenticación
- 14.6. Creación de *dashboard* y navegación
 - 14.6.1. Diseño y desarrollo de *dashboards* con elementos de inteligencia artificial
 - 14.6.1.1. Definición de *dashboards*
 - 14.6.1.2. Importancia de la IA en los *dashboards*
 - 14.6.1.3. Herramientas y tecnologías para el diseño de *dashboards*
 - 14.6.2. Implementación de sistemas de navegación eficientes en aplicaciones móviles con IA
 - 14.6.2.1. Importancia de la navegación en aplicaciones móviles
 - 14.6.2.2. Integración de IA en la navegación
 - 14.6.3. Integración de funcionalidades de IA en el *dashboard* para mejorar la experiencia del usuario
 - 14.6.3.1. Consideraciones de diseño
 - 14.6.3.2. Desarrollo e integración
- 14.7. Creación de pantalla con listad
 - 14.7.1. Desarrollo de interfaces de usuario para pantallas con listados en aplicaciones móviles con IA
 - 14.7.1.1. Fundamentos del diseño de interfaces para aplicaciones móviles
 - Herramientas y tecnologías para desarrollar pantallas con listados
 - 14.7.1.2. Prácticas recomendadas para mejorar la usabilidad y accesibilidad
 - 14.7.2. Integración de algoritmos de recomendación y filtrado en la pantalla de listado
 - 14.7.2.1. Tipos de algoritmos de recomendación y su aplicación en aplicaciones móviles
 - 14.7.2.2. Implementación de funciones de filtrado basadas en preferencias del usuario
 - 14.7.2.3. Consideraciones éticas en la implementación de algoritmos de IA en pantallas de listado

- 14.7.3. Uso de patrones de diseño para la presentación efectiva de datos en el listado
- 14.7.4. Estrategias para la carga eficiente de datos en tiempo real en la pantalla con listado
- 14.8. Creación de pantalla de detalle
 - 14.8.1. Diseño y desarrollo de interfaces de usuario detalladas para la presentación de información específica
 - 14.8.2. Integración de funcionalidades de IA para enriquecer la pantalla de detalle
 - 14.8.3. Implementación de interacciones y animaciones en la pantalla de detalle
 - 14.8.4. Estrategias para la optimización del rendimiento en la carga y visualización de detalles en aplicaciones móviles con IA
- 14.9. Creación de pantalla de *settings*
 - 14.9.1. Desarrollo de interfaces de usuario para configuración y ajustes en aplicaciones móviles con IA
 - 14.9.1.1. Diseño intuitivo
 - 14.9.1.2. Organización de opciones
 - 14.9.1.3. Consideraciones de accesibilidad
 - 14.9.2. Integración de ajustes personalizados relacionados con componentes de Inteligencia Artificial
 - 14.9.2.1. Identificación de ajustes relevantes
 - 14.9.2.2. Personalización avanzada
 - 14.9.2.3. Compatibilidad con múltiples dispositivos
 - 14.9.3. Implementación de opciones de personalización y preferencias en la pantalla de configuración
 - 14.9.3.1. Selección de opciones personalizadas
 - 14.9.3.2. Persistencia de ajustes
 - 14.9.3.3. Sincronización en la nube
 - 14.9.4. Estrategias para la usabilidad y claridad en la presentación de opciones en la pantalla de *settings*
 - 14.9.4.1. Jerarquía visual
 - 14.9.4.2. Lenguaje claro y conciso
 - 14.9.4.3. Pruebas de usabilidad
- 14.10. Crear Iconos, splash y recursos gráficos para tu App con IA
 - 14.10.1. Diseño y creación de iconos atractivos para representar la aplicación móvil con IA
 - 14.10.1.1. Comprensión de los principios del diseño de iconos
 - 14.10.1.2. Uso de herramientas de diseño gráfico
 - 14.10.1.3. Otras herramientas de diseño gráfico
 - 14.10.1.4. Coherencia con la identidad visual de la marca
 - 14.10.2. Desarrollo de pantallas de inicio (splash) con elementos visuales impactantes
 - 14.10.2.1. Diseñar una pantalla de inicio que refleje la identidad de la aplicación
 - 14.10.2.2. Utilizar animaciones sutiles para mejorar la experiencia del usuario
 - 14.10.2.3. Optimizar el tiempo de carga para una experiencia fluida
 - 14.10.3. Selección y adaptación de recursos gráficos que mejoren la estética de la aplicación móvil
 - 14.10.3.1. Elegir imágenes y gráficos de alta calidad que complementen el diseño general
 - 14.10.3.2. Adaptar los recursos gráficos para que se ajusten al tema y estilo de la aplicación
 - 14.10.3.3. Considerar la accesibilidad y la legibilidad al seleccionar recursos visuales
 - 14.10.4. Estrategias para la consistencia y branding visual en los elementos gráficos de la aplicación con IA
 - 14.10.4.1. Establecer guías de diseño que definan el estilo visual de la aplicación
 - 14.10.4.2. Mantener la coherencia en la paleta de colores, tipografía y estilos de diseño
 - 14.10.4.3. Incorporar elementos de branding de manera sutil pero efectiva

Módulo 15. Inteligencia Artificial para QA Testing

- 15.1. Ciclo de vida de *testing*
 - 15.1.1. Descripción y comprensión del ciclo de vida de *testing* en el desarrollo de software
 - 15.1.2. Fases del ciclo de vida de *testing* y su importancia en el aseguramiento de la calidad
 - 15.1.3. Integración de la Inteligencia Artificial en diferentes etapas del ciclo de vida de *testing*
 - 15.1.4. Estrategias para la mejora continua del ciclo de vida de *testing* mediante el uso de IA
- 15.2. Test Cases y detección de bugs
 - 15.2.1. Diseño y escritura efectiva de casos de prueba en el contexto de QA testing
 - 15.2.1.1. Identificar los requisitos de prueba y escenarios de prueba relevantes
 - 15.2.1.2. Utilizar técnicas como la partición de equivalencia y análisis de valores límite
 - 15.2.1.3. Escribir casos de prueba claros, concisos y fácilmente reproducibles
 - 15.2.2. Identificación de bugs y errores durante la ejecución de casos de prueba
 - 15.2.2.1. Establecer métricas para evaluar la efectividad de las pruebas
 - 15.2.2.2. Registrar y documentar adecuadamente los errores encontrados durante la ejecución de casos de prueba
 - 15.2.2.3. Implementar estrategias para reproducir y aislar bugs
 - 15.2.2.4. Colaborar eficazmente con el equipo de desarrollo para resolver problemas identificados
 - 15.2.3. Aplicación de técnicas de detección temprana de bugs mediante análisis estático
 - 15.2.3.1. Comprender el análisis estático y su importancia en la detección temprana de errores
 - 15.2.3.2. Uso de herramientas de análisis estático para identificar posibles problemas en el código
 - 15.2.3.3. Interpretación de los resultados del análisis estático y priorización de los problemas encontrados
 - 15.2.3.4. Integrar el análisis estático en el proceso de desarrollo de software para una detección proactiva de bugs
 - 15.2.4. Uso de herramientas de Inteligencia Artificial para la identificación automática de bugs en test cases
 - 15.2.4.1. Explorar diferentes enfoques de IA aplicados a QA Testing
 - 15.2.4.2. Evaluar herramientas y plataformas de IA disponibles para la identificación automática de bugs
 - 15.2.4.3. Integración de herramientas de IA en el flujo de trabajo de pruebas existente
 - 15.2.4.4. Análisis de casos de estudio y ejemplos de éxito en la aplicación de IA en QA Testing
- 15.3. Tipos de testing
 - 15.3.1. Exploración de diferentes tipos de testing en el ámbito de QA
 - 15.3.2. Pruebas unitarias, integración, funcionales, y de aceptación: características y aplicaciones
 - 15.3.2.1. Pruebas unitarias
 - 15.3.2.2. Pruebas de integración
 - 15.3.2.3. Pruebas funcionales
 - 15.3.2.4. Pruebas de aceptación
 - 15.3.3. Estrategias para la selección y combinación adecuada de tipos de testing en proyectos con ChatGPT
 - 15.3.4. Adaptación de tipos de testing convencionales a proyectos con ChatGPT
- 15.4. Crear un plan de pruebas
 - 15.4.1. Diseño y estructuración de un plan de pruebas integral
 - 15.4.2. Identificación de requisitos y escenarios de prueba en proyectos con IA
 - 15.4.3. Estrategias para la planificación de pruebas manuales y automatizadas
 - 15.4.4. Evaluación y ajuste continuo del plan de pruebas en función del desarrollo del proyecto
- 15.5. Detección y reportar bugs con IA
 - 15.5.1. Implementación de técnicas de detección automática de bugs mediante algoritmos de aprendizaje automático
 - 15.5.2. Uso de ChatGPT para el análisis dinámico de código en busca de posibles errores
 - 15.5.3. Estrategias para la generación automática de informes detallados sobre bugs detectados usando ChatGPT
 - 15.5.4. Colaboración efectiva entre equipos de desarrollo y QA en la gestión de bugs identificados por IA

- 15.6. Creación de pruebas automatizadas con IA
 - 15.6.1. Desarrollo de scripts de prueba automatizados para proyectos usando ChatGPT
 - 15.6.1.1. Concepto y beneficios de los scripts de prueba automatizados
Introducción a ChatGPT
 - 15.6.1.2. Proceso de desarrollo de scripts
 - 15.6.2. Integración de herramientas de automatización de pruebas basadas en IA
 - 15.6.2.1. Panorama de Herramientas de Automatización de Pruebas
 - 15.6.2.2. Ventajas de la integración de IA en herramientas de pruebas
 - 15.6.2.3. Selección de la Herramienta Adecuada
 - 15.6.2.4. Configuración y Personalización para el Entorno del Proyecto
 - 15.6.2.5. Implementación de ChatGPT en la herramienta seleccionada
 - 15.6.2.6. Ejemplo de Integración de ChatGPT con una Herramienta de Automatización de Pruebas
 - 15.6.3. Uso de ChatGPT para la generación dinámica de casos de prueba automatizados
 - 15.6.3.1. Generación Dinámica de Casos de Prueba con ChatGPT
 - 15.6.3.2. Implementación y Beneficios Prácticos
 - 15.6.4. Estrategias para la ejecución eficiente y mantenimiento de pruebas automatizadas en proyectos con IA
 - 15.6.4.1. Mantenimiento de Pruebas Automatizadas
 - 15.6.4.2. Estrategias Adicionales y Mejores Prácticas
- 15.7. API Testing
 - 15.7.1. Conceptos fundamentales de API testing y su importancia en QA
 - 15.7.1.1. Desarrollo
 - 15.7.2. Desarrollo de pruebas para la verificación de APIs en entornos usando ChatGPT
 - 15.7.2.1. Desarrollo
 - 15.7.3. Estrategias para la validación de datos y resultados en API testing con ChatGPT
 - 15.7.4. Uso de herramientas específicas para el testing de APIs en proyectos con Inteligencia Artificial
- 15.8. Herramientas de IA para Web testing
 - 15.8.1. Exploración de herramientas de inteligencia artificial para la automatización de pruebas en entornos web
 - 15.8.1.1. Análisis de herramientas populares
 - 15.8.1.2. Casos de uso y ejemplos prácticos
 - 15.8.2. Integración de tecnologías de reconocimiento de elementos y análisis visual en web testing
 - 15.8.2.1. Tecnologías de reconocimiento de elementos
 - 15.8.2.2. Análisis visual
 - 15.8.2.3. Integración en el proceso de *testing*
 - 15.8.2.4. Casos de uso y ejemplos prácticos
 - 15.8.3. Estrategias para la detección automática de cambios y problemas de rendimiento en aplicaciones web usando ChatGPT
 - 15.8.3.1. Introducción a ChatGPT
 - 15.8.3.2. Entrenamiento del modelo
 - 15.8.3.3. Detección de cambios automáticos
 - 15.8.3.4. Detección de problemas de rendimiento
 - 15.8.3.5. Integración en el proceso de *testing*
 - 15.8.3.6. Caso de uso y ejemplo práctico
 - 15.8.4. Evaluación de herramientas específicas para la mejora de la eficiencia en el web testing con IA
 - 15.8.4.1. Selección de herramientas para evaluación
 - 15.8.4.2. Criterios de evaluación
 - 15.8.4.3. Proceso de evaluación
 - 15.8.4.4. Resultados y conclusiones
- 15.9. Mobile Testing mediante IA
 - 15.9.1. Desarrollo de estrategias de testing para aplicaciones móviles con componentes de Inteligencia Artificial
 - 15.9.1.1. Identificación de casos de uso críticos
 - 15.9.1.2. Selección de herramientas de IA adecuadas
 - 15.9.1.3. Generación de datos de prueba
 - 15.9.1.4. Diseño de casos de prueba adaptativos
 - 15.9.1.5. Monitoreo de la efectividad de las pruebas
 - 15.9.2. Integración de herramientas de testing específicas para plataformas móviles basadas en IA
 - 15.9.2.1. Evaluación de herramientas disponibles Integración con el entorno de desarrollo
 - 15.9.2.2. Automatización de pruebas móviles
 - 15.9.2.3. Análisis de resultados de pruebas Adaptación y mejora continua

- 15.9.3. Uso de ChatGPT para la detección de problemas en el rendimiento de aplicaciones móviles
 - 15.9.3.1. Generación de informes de errores automatizados
 - 15.9.3.2. Diagnóstico de problemas de manera proactiva
 - 15.9.3.3. Generación de recomendaciones de mejora
 - 15.9.3.4. Automatización de acciones correctivas Colaboración y comunicación mejoradas
- 15.9.4. Estrategias para la validación de interfaces y funciones específicas de aplicaciones móviles mediante IA
 - 15.9.4.1. Reconocimiento de patrones en interfaces de usuario
 - 15.9.4.2. Pruebas de usabilidad automatizadas
 - 15.9.4.3. Validación de funciones específicas mediante procesamiento de lenguaje natural
 - 15.9.4.4. Pruebas de integración de funciones
 - 15.9.4.5. Retroalimentación basada en aprendizaje automático
- 15.10. Herramientas de QA con IA
 - 15.10.1. Exploración de herramientas y plataformas de QA que incorporan funcionalidades de Inteligencia Artificial
 - 15.10.1.1. Testim
 - 15.10.1.2. Mabl
 - 15.10.1.3. AppliTools
 - 15.10.1.4. Funcionalidades comunes de estas herramientas
 - 15.10.2. Evaluación de herramientas para la gestión y ejecución eficiente de pruebas en proyectos con IA
 - 15.10.3. Uso de ChatGPT para la generación y optimización de casos de prueba
 - 15.10.3.1. Generación automática de casos de prueba
 - 15.10.3.2. Optimización de casos de prueba existentes
 - 15.10.3.3. Adaptación a cambios en el software Integración con herramientas de gestión de pruebas
 - 15.10.4. Estrategias para la selección y adopción efectiva de herramientas de QA con capacidades de IA



“Elevarás de manera potencial tus competencias para el uso de ChatGPT en la detección de problemas en el rendimiento de aplicaciones móviles”

04

Objetivos docentes

Este Máster de Formación Permanente tiene como meta principal dotar a los profesionales de las competencias necesarias para implementar soluciones basadas en Inteligencia Artificial en entornos tecnológicos avanzados. A través de un análisis profundo de la historia de la IA y las estrategias para su implantación, el alumnado adquirirá habilidades para aplicar estas tecnologías de manera efectiva. Además, se abordará el análisis de datos y su integración en sistemas inteligentes, capacitando a los profesionales para diseñar y gestionar proyectos que optimicen el rendimiento y la eficiencia en el desarrollo de aplicaciones inteligentes en diversos sectores.



“

Incorporarás a tu práctica diaria las últimas tendencias en el uso de Inteligencia Artificial para QA Testing, mejorando la calidad del software”



Objetivos generales

- ♦ Desarrollar una comprensión integral sobre los fundamentos de la Inteligencia Artificial y su evolución tecnológica
- ♦ Capacitar en el manejo, preprocesamiento y transformación de datos para su integración en sistemas de IA
- ♦ Proporcionar habilidades en la creación y optimización de algoritmos en IA, con enfoque en la complejidad y eficiencia
- ♦ Fomentar la capacidad de implementar sistemas inteligentes y soluciones de aprendizaje automático en diversos contextos
- ♦ Profundizar en el uso de redes neuronales profundas y técnicas de *Deep Learning* para la resolución de problemas complejos
- ♦ Ahondar en el uso de herramientas como TensorFlow para personalizar y entrenar modelos de IA en aplicaciones reales
- ♦ Desarrollar competencias en el uso de IA para mejorar la productividad y optimizar el desarrollo de software
- ♦ Capacitar en la implementación de soluciones de IA para aplicaciones web y móviles, incluyendo QA *Testing* y arquitectura de software





Objetivos específicos

Módulo 1. Fundamentos de la Inteligencia Artificial

- ♦ Examinar los hitos históricos, las bases conceptuales y las tecnologías que sustentan la Inteligencia Artificial para comprender su evolución dentro de la informática actual
- ♦ Interpretar modelos clásicos de IA, como juegos, redes neuronales, algoritmos genéticos y sistemas expertos, desde una perspectiva computacional y algorítmica
- ♦ Relacionar agentes inteligentes, ontologías, web semántica y asistentes virtuales con arquitecturas de software orientadas a automatización e interoperabilidad
- ♦ Identificar criterios de implantación responsable y proyección tecnológica de la IA para orientar desarrollos sólidos en entornos digitales complejos

Módulo 2. El Dato en la Inteligencia Artificial

- ♦ Analizar el dato como activo central en sistemas inteligentes, vinculando su estructura, origen y tratamiento con procesos computacionales escalables
- ♦ Aplicar técnicas de análisis, visualización y depuración para transformar *datasets* heterogéneos en insumos fiables para el diseño de modelos de IA
- ♦ Detectar problemas de sesgo, desbalanceo, ruido y dimensionalidad que afecten el rendimiento de soluciones basadas en datos
- ♦ Diferenciar enfoques supervisados y no supervisados, integrando buenas prácticas de modelado y selección de herramientas en flujos de trabajo de informática

Módulo 3. Minería de Datos. Selección, Preprocesamiento y Transformación

- ♦ Utilizar fundamentos de inferencia estadística y análisis exploratorio para descubrir patrones significativos en grandes volúmenes de datos
- ♦ Aplicar procesos de integración, limpieza, normalización y transformación que mejoren la calidad computacional de los datos de entrada
- ♦ Tratar valores perdidos, ruido y alta dimensionalidad mediante técnicas que incrementen la robustez y precisión de los algoritmos
- ♦ Seleccionar atributos, instancias y estrategias de preprocesamiento adecuadas para sistemas de minería de datos en escenarios *Big Data*

Módulo 4. Sistemas Inteligentes

- ♦ Comprender el funcionamiento de los sistemas inteligentes a partir de la teoría de agentes, sus arquitecturas y sus mecanismos de razonamiento
- ♦ Estructurar información y conocimiento mediante ontologías, metadatos y modelos semánticos aplicables a software inteligente
- ♦ Incorporar principios de IA explicable para mejorar la trazabilidad, interpretabilidad y confianza técnica en sistemas computacionales avanzados
- ♦ Valorar el papel de razonadores, sistemas basados en conocimiento y sistemas expertos en la resolución automatizada de problemas complejos

Módulo 5. Aprendizaje Automático

- ♦ Distinguir los principales paradigmas de aprendizaje automático y su aplicación en tareas de clasificación, regresión, agrupamiento y asociación
- ♦ Comparar algoritmos, métricas y estrategias de combinación de modelos según el comportamiento esperado y las restricciones del problema
- ♦ Analizar modelos probabilísticos, redes neuronales y enfoques multclasificador para ampliar la capacidad predictiva de soluciones software
- ♦ Explorar arquitecturas profundas y librerías especializadas para desarrollar sistemas inteligentes con mayor precisión, adaptabilidad y rendimiento

Módulo 6. Las Redes Neuronales, Base de *Deep Learn*

- ♦ Comprender la lógica interna de las redes neuronales, desde su inspiración biológica hasta su formalización matemática y computacional
- ♦ Construir arquitecturas básicas identificando el papel de capas, operaciones, funciones de activación y procesos de propagación
- ♦ Configurar procesos de entrenamiento mediante optimizadores, funciones de pérdida y métricas ajustadas a distintos tipos de problema
- ♦ Implementar perceptrones multicapa con Keras y ajustar hiperparámetros clave para mejorar su comportamiento en escenarios reales de informática

Módulo 7. Entrenamiento de Redes Neuronales Profundas

- ♦ Analizar los desafíos técnicos del entrenamiento profundo, incluyendo gradientes inestables, convergencia deficiente y sobreajuste
- ♦ Emplear técnicas de inicialización, optimización y programación de tasa de aprendizaje para estabilizar el desempeño de redes profundas
- ♦ Aprovechar transfer learning y data augmentation como estrategias para acelerar desarrollos y mejorar generalización en modelos complejos
- ♦ Aplicar criterios de validación, regularización y contraste experimental que respalden configuraciones de entrenamiento más eficientes y sostenibles

Módulo 8. Personalización de Modelos y Entrenamientos con TensorFlow

- ♦ Utilizar TensorFlow como base para construir modelos personalizables con control preciso sobre funciones, gráficos y procesos de entrenamiento
- ♦ Integrar TensorFlow con NumPy, tf.data y TFRecord para diseñar pipelines eficientes de carga, transformación y consumo de datos
- ♦ Desarrollar modelos generativos, capas de preprocesamiento y flujos personalizados que respondan a necesidades específicas de software inteligente
- ♦ Implementar una aplicación práctica de *Deep Learning* con TensorFlow, conectando entrenamiento, despliegue funcional y predicción operativa

Módulo 9. Procesamiento del Lenguaje Natural (NLP)

- ♦ Comprender técnicas de NLP basadas en RNN, atención y Transformers para abordar generación, clasificación y traducción automática de texto
- ♦ Preparar corpus de entrenamiento mediante procesos de limpieza, transformación y estructuración adecuados para modelos secuenciales
- ♦ Evaluar bibliotecas y arquitecturas de NLP según rendimiento, flexibilidad e integración en proyectos de desarrollo de software
- ♦ Diseñar una aplicación que combine distintos enfoques de procesamiento lingüístico para resolver tareas avanzadas de interacción hombre - máquina

Módulo 10. Computación Bioinspirada

- ♦ Examinar la computación bioinspirada como enfoque de optimización aplicable a problemas de búsqueda, ajuste y exploración de soluciones complejas
- ♦ Analizar algoritmos de colonia de hormigas, partículas, genética y evolución diferencial desde una perspectiva de implementación y rendimiento
- ♦ Aplicar estrategias multiobjetivo y modelos neuro-evolutivos para resolver escenarios donde conviven restricciones, conflicto y alta complejidad
- ♦ Valorar el potencial de los modelos híbridos neuro - evolutivos como base para sistemas adaptativos de nueva generación en informática avanzada

Módulo 11. Mejora de la Productividad en Desarrollo de Software con Inteligencia Artificial

- ♦ Optimizar entornos de desarrollo incorporando herramientas de IA que reduzcan fricción técnica, mejoren automatización y aceleren ciclos de entrega
- ♦ Aplicar asistentes como ChatGPT o GitHub Copilot para refactorización, documentación, generación de pruebas y comprensión de código legado
- ♦ Integrar IA en repositorios, bases de datos, traducción entre lenguajes y flujos colaborativos para elevar productividad y consistencia técnica
- ♦ Evaluar herramientas de apoyo al desarrollo según contexto, complejidad del proyecto, compatibilidad y retorno operativo dentro del ciclo de software

Módulo 12. Arquitectura del software con Inteligencia Artificial

- ♦ Diseñar arquitecturas de software con IA priorizando escalabilidad, rendimiento, mantenibilidad y seguridad en entornos de alta exigencia
- ♦ Aplicar principios de *clean architecture*, patrones de diseño y microservicios para estructurar sistemas inteligentes desacoplados y extensibles
- ♦ Seleccionar estructuras de datos, algoritmos y tecnologías de almacenamiento alineadas con productos que procesan grandes volúmenes de información
- ♦ Incorporar prácticas de desarrollo seguro y observabilidad continua para proteger aplicaciones web con componentes de IA en producción

Módulo 13. Proyectos web con Inteligencia Artificial

- ♦ Configurar entornos, *workspaces* y flujos colaborativos que faciliten el desarrollo web con componentes de Inteligencia Artificial
- ♦ Integrar modelos y servicios de IA en frontend, backend y bases de datos, garantizando rendimiento, seguridad y coherencia arquitectónica
- ♦ Implementar estrategias de despliegue continuo, computación en nube y mantenimiento evolutivo para aplicaciones web inteligentes
- ♦ Desarrollar proyectos en entornos LAMP y MEVN incorporando procesamiento de datos, automatización y capacidades predictivas

Módulo 14. Aplicaciones Móviles con Inteligencia Artificial

- ♦ Preparar entornos de desarrollo móvil capaces de integrar bibliotecas, servicios y modelos de IA en aplicaciones multiplataforma o nativas
- ♦ Estructurar aplicaciones con *clean architecture*, repositorios y fuentes de datos que favorezcan modularidad, escalabilidad y mantenimiento
- ♦ Desarrollar pantallas, *dashboards*, listados y configuraciones que incorporen recomendaciones, personalización y procesamiento inteligente en tiempo real
- ♦ Diseñar recursos gráficos, autenticación, navegación y sincronización con Firebase para consolidar experiencias móviles funcionales y robustas





Módulo 15. Inteligencia Artificial para QA Testing

- ♦ Comprender el ciclo de vida del *testing* y su evolución mediante técnicas de Inteligencia Artificial aplicadas al aseguramiento de la calidad
- ♦ Diseñar casos de prueba, planes de validación y estrategias de detección de bugs con apoyo de análisis estático, automatización y modelos inteligentes
- ♦ Integrar IA en pruebas web, móviles y de APIs para incrementar cobertura, velocidad de ejecución y capacidad de detección temprana
- ♦ Seleccionar herramientas de QA con IA que fortalezcan generación de pruebas, análisis de resultados y mantenimiento continuo del software

“

Perfeccionarás tus competencias en clean architecture, optimizando la escalabilidad de tus proyectos de manera estructurada”

05

Salidas profesionales

El dominio de las técnicas avanzadas en Inteligencia Artificial en programación que adquirirá el alumno en esta novedosa titulación universitario abre un abanico de oportunidades en sectores tecnológicos de vanguardia. En primer lugar, profesionales capacitados en IA podrán acceder a roles clave como desarrollador de sistemas inteligentes, ingeniero de datos o especialista en aprendizaje automático. Por otro lado, esta capacitación especializará para integrarse en empresas innovadoras que lideran la transformación digital, así como en proyectos que requieren soluciones precisas y eficaces. Como resultado, los profesionales se posicionarán como líderes en creación de nuevas soluciones tecnológicas.



“

Te convertirás en un referente potencial en el diseño, implementación y mejora de soluciones como desarrollador de sistemas inteligentes”

Perfil del egresado

El egresado contará con competencias avanzadas para aplicar la Inteligencia Artificial en el desarrollo de soluciones tecnológicas innovadoras. De hecho, podrá diseñar modelos entrenados para tareas específicas, automatizar procesos complejos y optimizar sistemas mediante técnicas de aprendizaje automático. Del mismo modo, será capaz de integrar arquitecturas eficientes, trabajar con datos estructurados y no estructurados, y adaptar algoritmos a distintos entornos de desarrollo. Así, estará preparado para asumir funciones estratégicas en sectores que demandan automatización inteligente, análisis predictivo y toma de decisiones basada en datos. En definitiva, su perfil responderá a los retos tecnológicos del presente y futuro.

Asumirás roles especializados en el tratamiento de datos estructurados y no estructurados, dominando técnicas avanzadas para su procesamiento.

- ♦ **Pensamiento crítico:** Evaluar de manera objetiva datos, algoritmos y resultados, permitiendo tomar decisiones fundamentadas en evidencia y adaptadas a contextos cambiantes
- ♦ **Resolución de problemas complejos:** Identificar desafíos técnicos en entornos dinámicos y proponer soluciones eficientes a través de enfoques innovadores y tecnológicos
- ♦ **Adaptabilidad tecnológica:** Incorporar nuevas herramientas, lenguajes de programación y metodologías, respondiendo con agilidad a los avances del entorno digital
- ♦ **Comunicación técnica efectiva:** Transmitir ideas, resultados y propuestas de forma clara y estructurada, tanto a públicos especializados como no técnicos, facilitando el trabajo colaborativo



Después de realizar el programa universitario, podrás desempeñar tus conocimientos y habilidades en los siguientes cargos:

- 1. Ingeniero de Inteligencia Artificial:** Encargado de diseñar, entrenar y optimizar modelos de IA aplicables a diversos sectores, integrando algoritmos avanzados para resolver problemas complejos.
- 2. Desarrollador de sistemas inteligentes:** Responsable de implementar soluciones automatizadas que simulan comportamientos humanos, empleando técnicas de *machine learning* y procesamiento de datos.
- 3. Arquitecto de soluciones con IA:** Dedicado a la definición de estructuras tecnológicas que incorporan IA, asegurando la escalabilidad, eficiencia y mantenimiento de los sistemas desarrollados.
- 4. Especialista en aprendizaje automático:** Encargado de crear modelos predictivos basados en datos, ajustando parámetros y evaluando su rendimiento para mejorar la toma de decisiones automatizadas.
- 5. Científico de datos:** Responsable de analizar grandes volúmenes de información para extraer patrones, construir modelos y generar conocimiento útil en procesos de negocio.
- 6. Ingeniero de datos:** Gestor en el mantenimiento de infraestructuras para la recolección, almacenamiento y procesamiento de datos necesarios para el entrenamiento de modelos de IA.
- 7. Desarrollador de aplicaciones con IA:** Dedicado a la integración de algoritmos inteligentes en aplicaciones móviles o web, mejorando la experiencia del usuario y automatizando funcionalidades.

- 8. Especialista en visión por computador:** Responsable de aplicar técnicas de *Deep Learning* y redes neuronales para interpretar imágenes y videos, desarrollando sistemas de reconocimiento visual.
- 9. Ingeniero de procesamiento del lenguaje natural (NLP):** Gestor en la interpretación automática de textos y lenguaje hablado, permitiendo la interacción entre humanos y máquinas en lenguaje natural.
- 10. Consultor en transformación digital con IA:** Asesor de organizaciones en la adopción de tecnologías inteligentes, optimizando procesos mediante estrategias basadas en Inteligencia Artificial.



Te especializarás como ingeniero de Inteligencia Artificial en un área clave de la innovación tecnológica, con una titulación que integra vanguardia y excelencia”

06

Licencias de software incluidas

TECH es referencia en el mundo universitario por combinar la última tecnología con las metodologías docentes para potenciar el proceso de enseñanza-aprendizaje. Para ello, ha establecido una red de alianzas que le permite tener acceso a las herramientas de software más avanzadas del mundo profesional.



“

Al matricularte recibirás, de forma completamente gratuita, las credenciales de uso académico de las siguientes aplicaciones de software profesional”

TECH ha establecido una red de alianzas profesionales en la que se encuentran los principales proveedores de software aplicado a las diferentes áreas profesionales. Estas alianzas permiten a TECH tener acceso al uso de centenares de aplicaciones informáticas y licencias de software para acercarlas a sus estudiantes.

Las licencias de software para uso académico permitirán a los estudiantes utilizar las aplicaciones informáticas más avanzadas en su área profesional, de modo que podrán conocerlas y aprender su dominio sin tener que incurrir en costes. TECH se hará cargo del procedimiento de contratación para que los alumnos puedan utilizarlas de modo ilimitado durante el tiempo que estén estudiando el programa de Máster de Formación Permanente en Inteligencia Artificial en la Programación, y además lo podrán hacer de forma completamente gratuita.

TECH le dará acceso gratuito al uso de las siguientes aplicaciones de software:



Segments.ai

Segments.ai es una plataforma profesional valorada en **9.600 dólares**. Durante su inscripción en TECH Universidad, los profesionales accederán **sin costo** a esta solución, que revoluciona la gestión de proyectos complejos, mejorando la precisión, eficiencia y calidad de los procesos técnicos y creativos, permitiendo ejecutar tareas avanzadas en entornos profesionales con resultados consistentes y de alto nivel.

Acceder a **Segments.ai** ofrece un beneficio exclusivo: integrar una solución de referencia en proyectos complejos, optimizando tiempos, fortaleciendo la colaboración y garantizando resultados de impacto en la organización y manejo de información visual, todo dentro de un entorno profesional competitivo, preparado para desafíos técnicos y creativos, con capacidad de mejorar procesos y maximizar el valor de cada proyecto.

Flutterflow

FlutterFlow permite diseñar, prototipar y generar aplicaciones móviles y web de manera visual e intuitiva. Con un valor aproximado de **360 dólares**, TECH ofrece acceso totalmente **gratuito** durante el programa universitario.

Brinda un entorno simplificado que agiliza los procesos de producción, facilitando la creación de soluciones digitales funcionales y precisas. Posibilita la creación automática de códigos limpios, reutilizables y compatibles con sistemas Android e iOS. Permite diseñar aplicaciones modernas, eficientes y alineadas con las demandas del mercado tecnológico actual.

GitHub Education

GitHub Education es una plataforma colaborativa para el desarrollo de software, con herramientas profesionales de control de versiones y gestión de proyectos. Valorada en aproximadamente **120 euros** anuales, este instrumento se ofrecerá al alumnado **gratuitamente** durante el programa universitario de TECH. Esta plataforma permite gestionar repositorios, colaborar en equipo y automatizar flujos de trabajo en escenarios reales.

Este instrumento incluye GitHub Copilot, compatibilidad con múltiples IDE y acceso a recursos técnicos que amplifican las capacidades de desarrollo y revisión de código en distintos lenguajes. Sus funciones abarcan control de versiones, entornos de desarrollo integrados, tableros Kanban, seguimiento de *issues* y comunidades técnicas. Así, se convierte en un recurso esencial para fomentar la eficiencia, colaboración y calidad en proyectos de software.

Google Career Launchpad

Google Career Launchpad es una solución diseñada para desarrollar competencias digitales en tecnología y análisis de datos. Valorada en aproximadamente **5.000 dólares**, esta licencia se ofrece **gratis** durante todo el programa universitario de TECH, brindando acceso a laboratorios interactivos, contenidos especializados y certificaciones de alto reconocimiento dentro del sector, fortaleciendo la proyección del egresado en entornos tecnológicos altamente competitivos.

Esta plataforma combina capacitación técnica con casos prácticos reales, empleando herramientas avanzadas como BigQuery y la Inteligencia Artificial de Google. Así pues, el egresado TECH podrá practicar en entornos virtuales, trabajar con datos reales, interactuar con expertos, resolver retos del sector y prepararse para certificaciones oficiales reconocidas globalmente, reforzando habilidades en análisis de datos, *machine learning* y soluciones innovadoras basadas en IA.

AskYourDatabase

AskYourDatabase es una solución innovadora para analizar datos mediante lenguaje natural. Valorada en **150 dólares**, los egresados del programa universitario de TECH tendrán acceso **gratuito** a este instrumento. Gracias a esto, los profesionales podrán generar *insights* sin necesidad de escribir código. Esta plataforma potencia el trabajo colaborativo, facilitando la gestión de repositorios, la colaboración en equipo y la automatización de flujos de trabajo.

Entre sus funciones destacan consultas en lenguaje natural, visualización interactiva de *dashboards*, conectividad multiplataforma, colaboración con permisos personalizables y modelos predictivos con Inteligencia Artificial integrada. Además, este instrumento combina simplicidad y potencia, ofreciendo a los egresados herramientas esenciales para tomar decisiones basadas en datos en entornos profesionales competitivos.



Gracias a TECH podrás utilizar gratuitamente las mejores aplicaciones de software de tu área profesional”

07

Metodología de estudio

TECH es la primera universidad en el mundo que combina la metodología de los **case studies** con el **Relearning**, un sistema de aprendizaje 100% online basado en la reiteración dirigida.

Esta disruptiva estrategia pedagógica ha sido concebida para ofrecer a los profesionales la oportunidad de actualizar conocimientos y desarrollar competencias de un modo intenso y riguroso. Un modelo de aprendizaje que coloca al estudiante en el centro del proceso académico y le otorga todo el protagonismo, adaptándose a sus necesidades y dejando de lado las metodologías más convencionales.



“

TECH te prepara para afrontar nuevos retos en entornos inciertos y lograr el éxito en tu carrera”

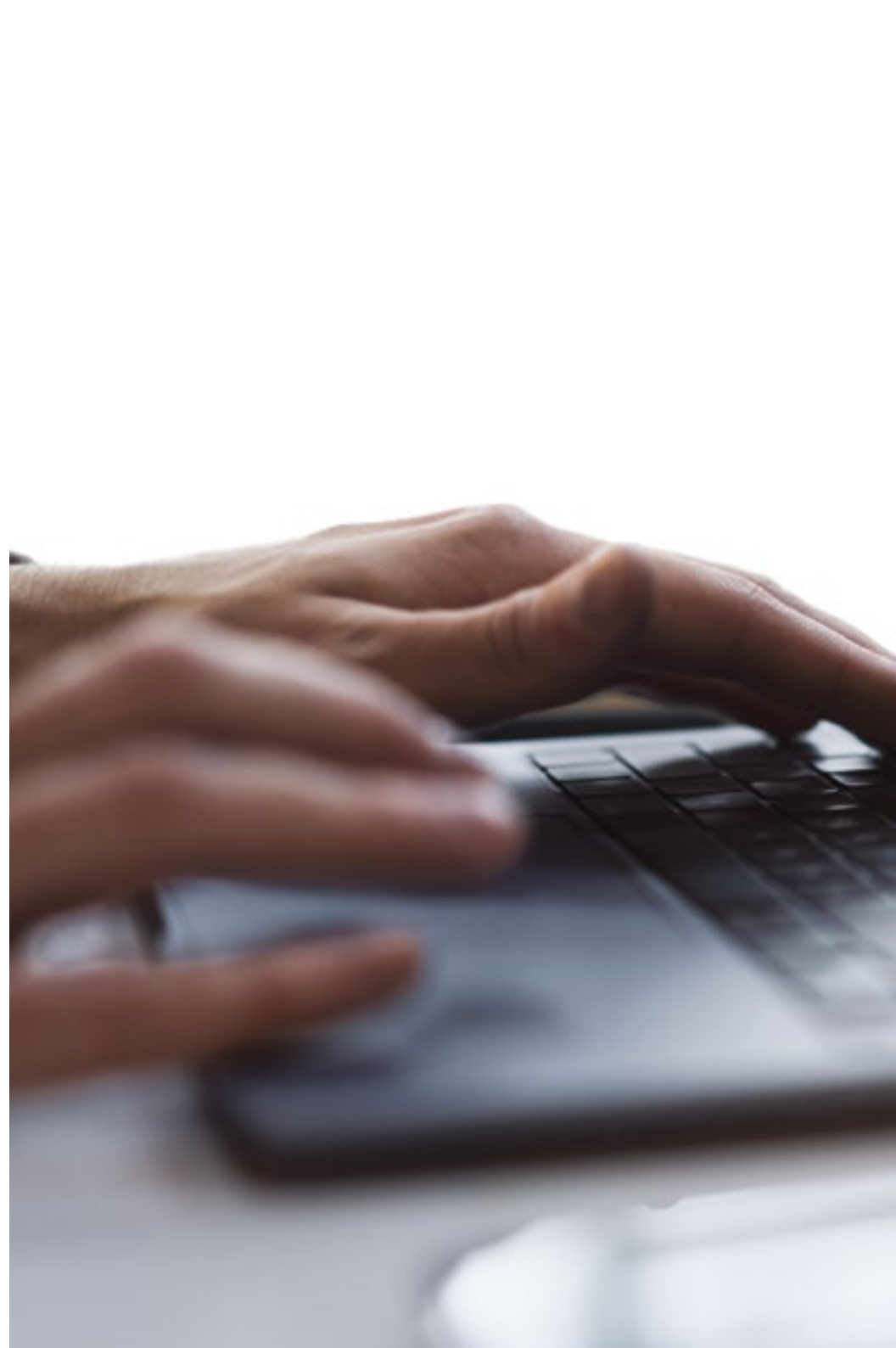
El alumno: la prioridad de todos los programas de TECH

En la metodología de estudios de TECH el alumno es el protagonista absoluto. Las herramientas pedagógicas de cada programa han sido seleccionadas teniendo en cuenta las demandas de tiempo, disponibilidad y rigor académico que, a día de hoy, no solo exigen los estudiantes sino los puestos más competitivos del mercado.

Con el modelo educativo asincrónico de TECH, es el alumno quien elige el tiempo que destina al estudio, cómo decide establecer sus rutinas y todo ello desde la comodidad del dispositivo electrónico de su preferencia. El alumno no tendrá que asistir a clases en vivo, a las que muchas veces no podrá acudir. Las actividades de aprendizaje las realizará cuando le venga bien. Siempre podrá decidir cuándo y desde dónde estudiar.

“

*En TECH NO tendrás clases en directo
(a las que luego nunca puedes asistir)”*



Los planes de estudios más exhaustivos a nivel internacional

TECH se caracteriza por ofrecer los itinerarios académicos más completos del entorno universitario. Esta exhaustividad se logra a través de la creación de temarios que no solo abarcan los conocimientos esenciales, sino también las innovaciones más recientes en cada área.

Al estar en constante actualización, estos programas permiten que los estudiantes se mantengan al día con los cambios del mercado y adquieran las habilidades más valoradas por los empleadores. De esta manera, quienes finalizan sus estudios en TECH reciben una preparación integral que les proporciona una ventaja competitiva notable para avanzar en sus carreras.

Y además, podrán hacerlo desde cualquier dispositivo, pc, tableta o smartphone.

“

El modelo de TECH es asincrónico, de modo que te permite estudiar con tu pc, tableta o tu smartphone donde quieras, cuando quieras y durante el tiempo que quieras”

Case studies o Método del caso

El método del caso ha sido el sistema de aprendizaje más utilizado por las mejores escuelas de negocios del mundo. Desarrollado en 1912 para que los estudiantes de Derecho no solo aprendiesen las leyes a base de contenidos teóricos, su función era también presentarles situaciones complejas reales. Así, podían tomar decisiones y emitir juicios de valor fundamentados sobre cómo resolverlas. En 1924 se estableció como método estándar de enseñanza en Harvard.

Con este modelo de enseñanza es el propio alumno quien va construyendo su competencia profesional a través de estrategias como el *Learning by doing* o el *Design Thinking*, utilizadas por otras instituciones de renombre como Yale o Stanford.

Este método, orientado a la acción, será aplicado a lo largo de todo el itinerario académico que el alumno emprenda junto a TECH. De ese modo se enfrentará a múltiples situaciones reales y deberá integrar conocimientos, investigar, argumentar y defender sus ideas y decisiones. Todo ello con la premisa de responder al cuestionamiento de cómo actuaría al posicionarse frente a eventos específicos de complejidad en su labor cotidiana.



Método Relearning

En TECH los *case studies* son potenciados con el mejor método de enseñanza 100% online: el *Relearning*.

Este método rompe con las técnicas tradicionales de enseñanza para poner al alumno en el centro de la ecuación, proveyéndole del mejor contenido en diferentes formatos. De esta forma, consigue repasar y reiterar los conceptos clave de cada materia y aprender a aplicarlos en un entorno real.

En esta misma línea, y de acuerdo a múltiples investigaciones científicas, la reiteración es la mejor manera de aprender. Por eso, TECH ofrece entre 8 y 16 repeticiones de cada concepto clave dentro de una misma lección, presentada de una manera diferente, con el objetivo de asegurar que el conocimiento sea completamente afianzado durante el proceso de estudio.

El Relearning te permitirá aprender con menos esfuerzo y más rendimiento, implicándote más en tu especialización, desarrollando el espíritu crítico, la defensa de argumentos y el contraste de opiniones: una ecuación directa al éxito.



Un Campus Virtual 100% online con los mejores recursos didácticos

Para aplicar su metodología de forma eficaz, TECH se centra en proveer a los egresados de materiales didácticos en diferentes formatos: textos, vídeos interactivos, ilustraciones y mapas de conocimiento, entre otros. Todos ellos, diseñados por profesores cualificados que centran el trabajo en combinar casos reales con la resolución de situaciones complejas mediante simulación, el estudio de contextos aplicados a cada carrera profesional y el aprendizaje basado en la reiteración, a través de audios, presentaciones, animaciones, imágenes, etc.

Y es que las últimas evidencias científicas en el ámbito de las Neurociencias apuntan a la importancia de tener en cuenta el lugar y el contexto donde se accede a los contenidos antes de iniciar un nuevo aprendizaje. Poder ajustar esas variables de una manera personalizada favorece que las personas puedan recordar y almacenar en el hipocampo los conocimientos para retenerlos a largo plazo. Se trata de un modelo denominado *Neurocognitive context-dependent e-learning* que es aplicado de manera consciente en esta titulación universitaria.

Por otro lado, también en aras de favorecer al máximo el contacto mentor-alumno, se proporciona un amplio abanico de posibilidades de comunicación, tanto en tiempo real como en diferido (mensajería interna, foros de discusión, servicio de atención telefónica, email de contacto con secretaría técnica, chat y videoconferencia).

Asimismo, este completísimo Campus Virtual permitirá que el alumnado de TECH organice sus horarios de estudio de acuerdo con su disponibilidad personal o sus obligaciones laborales. De esa manera tendrá un control global de los contenidos académicos y sus herramientas didácticas, puestas en función de su acelerada actualización profesional.



La modalidad de estudios online de este programa te permitirá organizar tu tiempo y tu ritmo de aprendizaje, adaptándolo a tus horarios”

La eficacia del método se justifica con cuatro logros fundamentales:

1. Los alumnos que siguen este método no solo consiguen la asimilación de conceptos, sino un desarrollo de su capacidad mental, mediante ejercicios de evaluación de situaciones reales y aplicación de conocimientos.
2. El aprendizaje se concreta de una manera sólida en capacidades prácticas que permiten al alumno una mejor integración en el mundo real.
3. Se consigue una asimilación más sencilla y eficiente de las ideas y conceptos, gracias al planteamiento de situaciones que han surgido de la realidad.
4. La sensación de eficiencia del esfuerzo invertido se convierte en un estímulo muy importante para el alumnado, que se traduce en un interés mayor en los aprendizajes y un incremento del tiempo dedicado a trabajar en el curso.

La metodología universitaria mejor valorada por sus alumnos

Los resultados de este innovador modelo académico son constatables en los niveles de satisfacción global de los egresados de TECH.

La valoración de los estudiantes sobre la calidad docente, calidad de los materiales, estructura del curso y sus objetivos es excelente. No en valde, la institución se convirtió en la universidad mejor valorada por sus alumnos según el índice global score, obteniendo un 4,9 de 5.

Accede a los contenidos de estudio desde cualquier dispositivo con conexión a Internet (ordenador, tablet, smartphone) gracias a que TECH está al día de la vanguardia tecnológica y pedagógica.

Podrás aprender con las ventajas del acceso a entornos simulados de aprendizaje y el planteamiento de aprendizaje por observación, esto es, Learning from an expert.



Así, en este programa estarán disponibles los mejores materiales educativos, preparados a conciencia:



Material de estudio

Todos los contenidos didácticos son creados por los especialistas que van a impartir el curso, específicamente para él, de manera que el desarrollo didáctico sea realmente específico y concreto.

Estos contenidos son aplicados después al formato audiovisual que creará nuestra manera de trabajo online, con las técnicas más novedosas que nos permiten ofrecerte una gran calidad, en cada una de las piezas que pondremos a tu servicio.



Prácticas de habilidades y competencias

Realizarás actividades de desarrollo de competencias y habilidades específicas en cada área temática. Prácticas y dinámicas para adquirir y desarrollar las destrezas y habilidades que un especialista precisa desarrollar en el marco de la globalización que vivimos.



Resúmenes interactivos

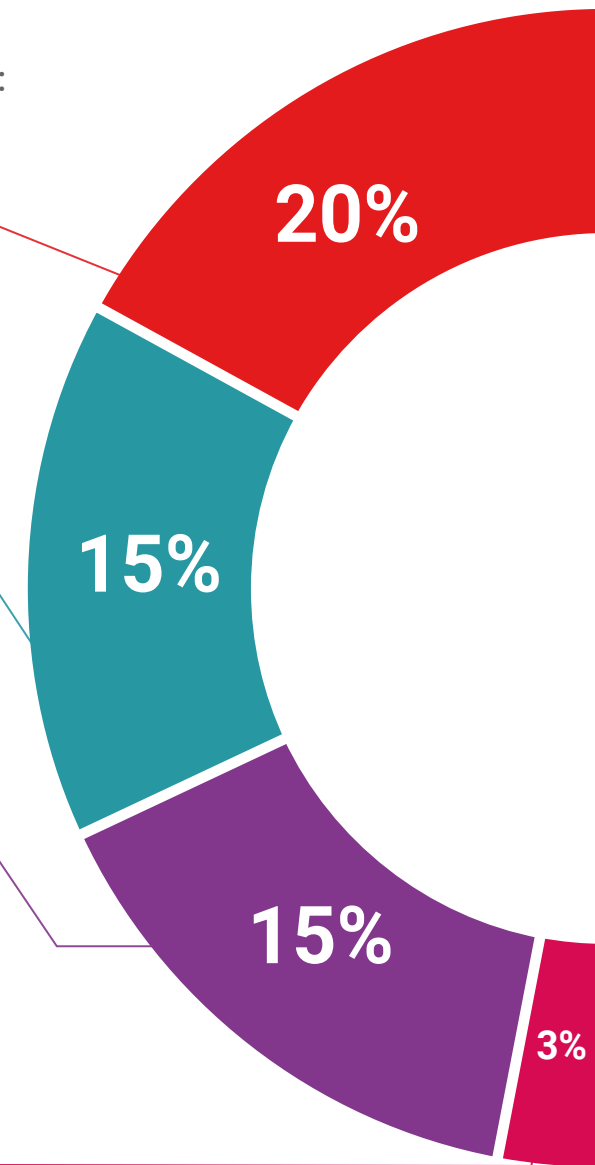
Presentamos los contenidos de manera atractiva y dinámica en píldoras multimedia que incluyen audio, vídeos, imágenes, esquemas y mapas conceptuales con el fin de afianzar el conocimiento.

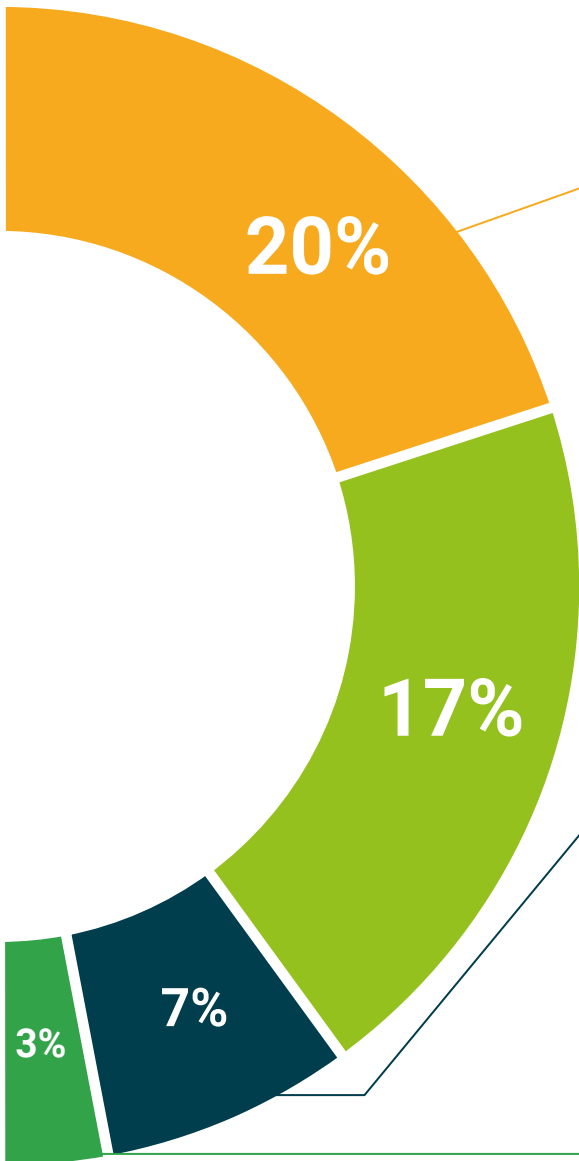
Este sistema exclusivo educativo para la presentación de contenidos multimedia fue premiado por Microsoft como "Caso de éxito en Europa".



Lecturas complementarias

Artículos recientes, documentos de consenso, guías internacionales... En nuestra biblioteca virtual tendrás acceso a todo lo que necesitas para completar tu capacitación.





Case Studies

Completarás una selección de los mejores *case studies* de la materia. Casos presentados, analizados y tutorizados por los mejores especialistas del panorama internacional.



Testing & Retesting

Evaluamos y reevaluamos periódicamente tu conocimiento a lo largo del programa. Lo hacemos sobre 3 de los 4 niveles de la Pirámide de Miller.



Clases magistrales

Existe evidencia científica sobre la utilidad de la observación de terceros expertos. El denominado *Learning from an expert* afianza el conocimiento y el recuerdo, y genera seguridad en nuestras futuras decisiones difíciles.



Guías rápidas de actuación

TECH ofrece los contenidos más relevantes del curso en forma de fichas o guías rápidas de actuación. Una manera sintética, práctica y eficaz de ayudar al estudiante a progresar en su aprendizaje.



08

Cuadro docente

En su compromiso con la enseñanza de élite, TECH ha seleccionado minuciosamente a los docentes encargados de la elaboración del plan de estudios de esta titulación. Por ello, el presente programa universitario cuenta con un experimentado claustro que posee un destacado bagaje en la aplicación de la Inteligencia Artificial en labores de Programación. De este modo, los alumnos de este Máster de Formación Permanente accederán a una experiencia académica de primer nivel, con una combinación exclusiva de conocimientos presentados en diversos soportes audiovisuales, para una integración de los conocimientos más eficaz y dinámica.



“

Ponte al día en las últimas tendencias en Inteligencia Artificial aplicada a la Programación de la mano de los mejores expertos en la materia”

Dirección



Dr. Peralta Martín-Palomino, Arturo

- ♦ CEO y CTO en Prometheus Global Solutions
- ♦ CTO en Korporate Technologies
- ♦ CTO en AI Shepherds GmbH
- ♦ Consultor y Asesor Estratégico Empresarial en Alliance Medical
- ♦ Director de Diseño y Desarrollo en DocPath
- ♦ Doctor en Ingeniería Informática por la Universidad de Castilla-La Mancha
- ♦ Doctor en Economía, Empresas y Finanzas por la Universidad Camilo José Cela
- ♦ Doctor en Psicología por la Universidad de Castilla-La Mancha
- ♦ Máster en Executive MBA por la Universidad Isabel I
- ♦ Máster en Dirección Comercial y Marketing por la Universidad Isabel I
- ♦ Máster Experto en Big Data por Formación Hadoop
- ♦ Máster en Tecnologías Informáticas Avanzadas por la Universidad de Castilla-La Mancha
- ♦ Miembro: Grupo de Investigación SMILE



D. Castellanos Herreros, Ricardo

- *Chief Technology Officer* en OWQLO
- Especialista en Ingeniería Informática de Sistemas y *Machine Learning Engineer*
- Consultor Técnico *Freelance*
- Desarrollador de Aplicaciones Móviles para eDreams, Fnac, Air Europa, Bankia, Cetelem, Banco Santander, Santillana, Groupón y Grupo Planeta
- Desarrollador de Páginas Web para Openbank y Banco Santander
- Ingeniero Técnico en Informática de Sistemas por la Universidad de Castilla la Mancha

09

Titulación

Este programa en Inteligencia Artificial en la Programación garantiza, además de la capacitación más rigurosa y actualizada, el acceso a un título de Máster de Formación Permanente expedido por TECH Universidad.



“

Supera con éxito este programa y recibe tu titulación universitaria sin desplazamientos ni farragosos trámites”

Este programa le permitirá obtener el título de **Máster de Formación Permanente en Inteligencia Artificial en la Programación** emitido por TECH Universidad.

TECH es una Universidad española oficial, que forma parte del Espacio Europeo de Educación Superior (EEES). Con un enfoque centrado en la excelencia académica y la calidad universitaria a través de la tecnología.

Este título propio contribuye de forma relevante al desarrollo de la educación continua y actualización del profesional, garantizándole la adquisición de las competencias en su área de conocimiento y aportándole un alto valor curricular universitario a su formación. Es 100% válido en todas las Oposiciones, Carrera Profesional y Bolsas de Trabajo de cualquier Comunidad Autónoma española.

Además, el riguroso sistema de garantía de calidad de TECH asegura que cada título otorgado cumpla con los más altos estándares académicos, brindándole al egresado la confianza y la credibilidad que necesita para destacarse en su carrera profesional.

Título: **Máster de Formación Permanente en Inteligencia Artificial en la Programación**

Modalidad: **online**

Duración: **12 meses**

Acreditación: **90 ECTS**



*Apostilla de La Haya. En caso de que el alumno solicite que su título en papel recabe la Apostilla de La Haya, TECH Universidad realizará las gestiones oportunas para su obtención, con un coste adicional.



Máster de Formación
Permanente
Inteligencia Artificial
en la Programación

- » Modalidad: online
- » Duración: 12 meses
- » Titulación: TECH Universidad
- » Acreditación: 90 ECTS
- » Horario: a tu ritmo
- » Exámenes: online

Máster de Formación Permanente

Inteligencia Artificial en la Programación