Máster Título Propio
Programación Cloud



tech universidad tecnológica



tech universidad tecnológica

Máster Título Propio
Programación Cloud

Modalidad: **Online**
Duración: **12 meses**
Titulación: **TECH Universidad Tecnológica**
Horas lectivas: **1.500 h.**
Acceso web: **www.techtitute.com/informatica/master/master-programacion-cloud**

Índice

01 Presentación — pág. 4

02 Objetivos — pág. 8

03 Competencias — pág. 14

04 Dirección del curso — pág. 18

05 Estructura y contenido — pág. 24

06 Metodología — pág. 36

07 Titulación — pág. 44

02

Objetivos

El diseño del programa de este Máster Título Propio permitirá al alumnado iniciar un proyecto en entornos *Cloud* en su propia *start-up* o en grandes compañías. Para ello, en esta titulación adquirirá los conocimientos especializados para determinar la *Cloud* adecuada, aprenderá a analizar los diferentes enfoques para la adopción de la nube y sus contextos, además de dominar las principales herramientas y proveedores en este campo tecnológico en alza. El contenido interactivo y los cases *simulation* permitirán afianzar los conocimientos en uno de los sectores tecnológicos con mayor potencial del presente.





Progresa en tu carrera profesional. Cada día las empresas demandan más ingenieros de Front-End o Back-End. Especialízate en un campo con amplias salidas laborales"

Objetivos generales

- Analizar los diferentes enfoques para la adopción de la nube y sus contextos
- Adquirir conocimiento especializado para determinar la *Cloud* adecuada
- Desarrollar una máquina virtual en Azure
- Establecer las fuentes de amenazas en el desarrollo de aplicaciones y las mejores prácticas a aplicar
- Evaluar las diferencias en las implementaciones concretas de diferentes vendedores de Cloud pública
- Determinar las diferentes tecnologías aplicadas a contenedores
- Identificar los aspectos clave en la adopción de una estrategia de adopción *Cloud-Native*
- Fundamentar y evaluar los lenguajes de programación más utilizados en *Big Data*, necesarios para el análisis y procesamiento del dato





Objetivos específicos

Módulo 1. Programación *Cloud*. Servicios en Azure, AWS y Google *Cloud*

- Generar conocimiento especializado sobre la nube y las diferencias con las soluciones locales tradicionales
- Adquirir vocabulario especializado fundamental en la nube. Dominar los términos utilizados por los diferentes proveedores
- Establecer los principales componentes de la nube y uso
- Determinar los proveedores en el mercado de la nube, sus fortalezas y debilidades, y aportaciones

Módulo 2. Programación de Arquitecturas en *Cloud Computing*

- Desarrollar conocimiento especializado sobre las bases de arquitectura
- Especializar al alumno en el conocimiento de infraestructuras *Cloud*
- Evaluar ventajas e inconvenientes de desplegar *On Premise* o en *Cloud*
- Determinar los requerimientos de infraestructura
- Identificar opciones de despliegue
- Capacitar para la puesta en producción de una infraestructura *Cloud*
- Diseñar y definir la operación y el mantenimiento de una arquitectura en *Cloud*

Módulo 3. Storage en *Cloud* Azure
- Examinar una máquina virtual en Azure
- Establecer los diferentes tipos de almacenamiento
- Evaluar las funciones en el *backup*
- Gestionar recursos de Azure
- Analizar los diferentes tipos de servicios
- Examinar los diferentes tipos de seguridad
- Generar redes virtuales
- Concretar las diferentes conexiones de red

Módulo 4. Entornos *Cloud*. Seguridad
- Identificar riesgos de un despliegue de infraestructura en *Cloud* pública
- Analizar los riesgos de seguridad en el desarrollo de aplicaciones
- Determinar los requerimientos de seguridad
- Desarrollar un plan de seguridad para un despliegue en *Cloud*
- Establecer directrices para un sistema de *logging* y monitorización
- Proponer acciones de respuesta ante incidentes

Módulo 5. Orquestación de contenedores: Kubernetes y Docker
- Desarrollar las bases de arquitectura y tecnología de contenedores
- Establecer las diferentes tecnologías aplicadas a contenedores
- Determinar los requerimientos de la infraestructura
- Examinar opciones de despliegue

Módulo 6. Programación de aplicaciones *Cloud* Nativa
- Presentar las tecnologías para desarrollo e integración continua
- Demostrar el funcionamiento de Kubernetes como orquestación de servicios
- Analizar las herramientas de observabilidad y seguridad en *Cloud-native*
- Evaluar las plataformas de despliegue
- Fundamentar las estrategias de gestión de datos en entornos *Cloud-native*
- Identificar técnicas comunes en desarrollos *Cloud-Native*

Módulo 7. Programación *Cloud*. Data Governance
- Generar conocimiento especializado sobre el manejo, estrategias, técnicas de procesamiento de datos
- Desarrollar estrategias de gobernanza de datos dirigidas a personas, procesos y herramientas
- Llevar a cabo la gobernanza de datos desde la ingesta hasta la preparación y uso
- Determinar técnicas para gobernar la transmisión de datos
- Establecer protección de datos para autenticación, seguridad, respaldo y monitores

Módulo 8. Programación *Cloud* en Tiempo real. *Streaming*
- Analizar el proceso de recolección, estructuración, procesado, análisis e interpretación de los datos *streaming*
- Desarrollar los principios del procesamiento en *streaming*, el contexto actual y los casos de uso actuales en el marco nacional
- Desarrollar fundamentos clave de estadística, *maching learning*, minería de datos y modelización predictiva para la comprensión del análisis y procesado de los datos
- Analizar los principales lenguajes de programación en Big Data
- Examinar los fundamentos de Apache Spark Streaming, Kafka Stream y Flink Stream



Módulo 9. Integración *Cloud* con Servicios Web. Tecnologías y Protocolos

- Evaluar el avance de las tecnologías y arquitecturas web para determinar la complejidad del sistema y en base a ello plantear una solución software
- Desarrollar proyectos distribuidos en *Cloud Computing* utilizando servicios web y distintos requisitos funcionales y de seguridad
- Analizar diferentes tecnologías de implementación de servicios web, identificando aquella que dé mejor soporte teniendo en cuenta el escenario del problema
- Evaluar la corrección en la implementación de un servicio web del lado servidor por medio del lanzamiento de peticiones desde diferentes tipologías de clientes web

Módulo 10. Programación *Cloud*. Gestión de proyectos y verificación del producto

- Conocer los escenarios y las aplicaciones en la gestión del ciclo de vida
- Gestionar proyectos como un proceso y determinar el modelo de organización
- Determinar los riesgos y los costes aplicando metodologías ágiles durante la fase de conceptualización o durante la ejecución del proyecto
- Dirigir y gestionar los proyectos con metodologías ágiles y la calidad de los proyectos *Cloud* aplicando diferentes metodologías

Los recursos multimedia y el sistema Relearning te ayudarán a cimentar tus conocimientos. Progresa en tu carrera profesional convirtiéndote en un experto en Programación Cloud”

03

Competencias

La estructura de este Máster Título Propio está confeccionada para que el profesional informático al que va dirigido sea capaz de llevar a cabo la programación *Cloud* en empresas que desean gestionar todos sus activos de manera ágil. Al finalizar esta titulación estará capacitado para manejar los diferentes proveedores existentes, a optimizar y diferenciar el uso de la tecnología de la virtualización, el *Cloud Computing* y los contenedores. El cuerpo docente lo guiará hacia la excelencia a lo largo de toda esta titulación.

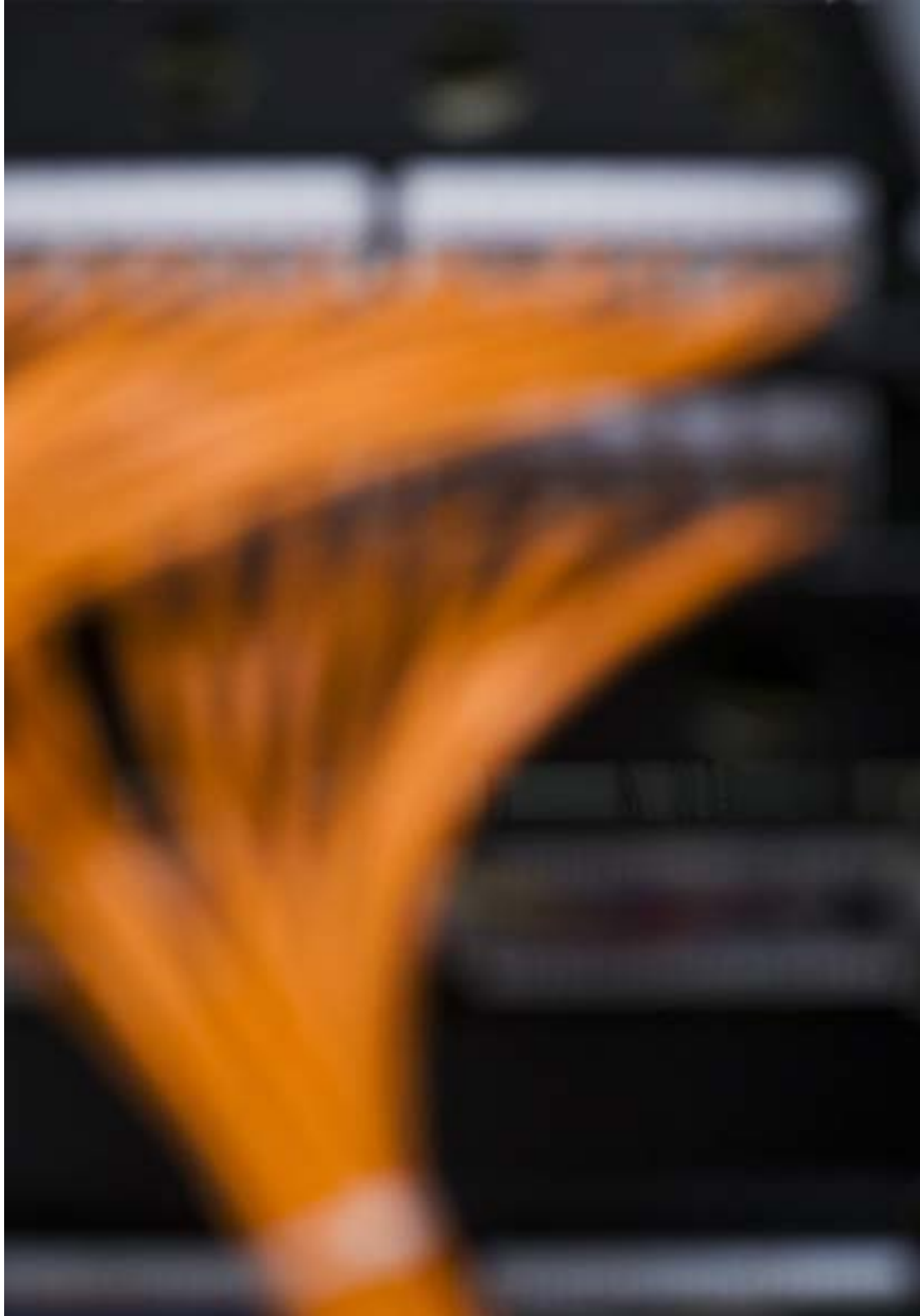




Aprende los fundamentos de la metodología DevOp y aplícalos en el desarrollo de productos digitales"

Competencias generales

- Analizar el proceso de transformación experimentado por las empresas que han adoptado la metodología DevOps
- Aprender los fundamentos de la metodología DevOps para aplicarlos en la gestión del proceso de implantación y desarrollo de productos digitales
- Dominar las diferentes técnicas de análisis de negocio existentes para el correcto desarrollo de la fase de planificación
- Manejar las técnicas de verificación y validación existentes para asegurar la calidad del producto desarrollado
- Establecer las diferencias existentes entre tecnologías de virtualización, *Cloud Computing* y contenedores para el óptimo uso de cada una de ellas
- Examinar las bases sobre las que se conforman los servicios en la nube para hacer un uso adecuado de estas herramientas en la empresa
- Conocer los proveedores y las características de los servicios ofrecidos en *Cloud Computing* para seleccionar aquellos que mejor se ajustan a las necesidades de la empresa





Competencias específicas

- Analizar los diferentes enfoques para la adopción de la nube y sus contextos
- Determinar las principales técnicas de procesado de datos
- Aumentar los conocimientos sobre las diferentes herramientas y su utilización en el ámbito de la gestión de proyectos
- Generar conocimiento especializado sobre la calidad del servicio y cómo lograr una calidad adecuada con su producto
- Fundamentar el uso concreto de una arquitectura de servicios para dar solución a un problema en un marco concreto
- Identificar las debilidades y amenazas de un sistema a fin de proponer una solución tecnológica que se soporte a la seguridad de un sistema
- Examinar los diferentes servicios proporcionados por los proveedores *Cloud* y fundamentar el uso de los mismos en un proyecto concreto
- Examinar el uso de contenedores y desarrollo con microservicios
- Identificar los servicios *Cloud* a desplegar para la ejecución de un plan de seguridad y la operativa necesaria para los mecanismos de prevención

04

Dirección del curso

En su máxima de ofrecer una educación de élite para todos, TECH selecciona rigurosamente a un equipo docente para que el alumnado adquiera un conocimiento sólido en la especialidad de programación Cloud. Por ello, el presente Máster Título Propio cuenta con un equipo altamente cualificado y con amplia experiencia en el sector. Gracias a sus conocimientos y cercanía con el ámbito profesional, el contenido de esta enseñanza acerca al profesional de la informática a las últimas novedades en esta tecnología y las necesidades actuales de las empresas en este campo.





Un potente cuadro docente te guiará durante los doce meses de duración de este Máster Título Propio. Haz clic e inscríbete"

Dirección



D. Bressel Gutiérrez-Ambrossi, Guillermo

- Administrador de Storage y red SAN en Experis IT (BBVA)
- Administrador de redes en IE Business School
- Graduado Superior en Administración de sistemas y redes informáticas en ASIR
- Curso Ethical Hacking en OpenWebinar
- Curso Powershel en OpenWebinar

Profesores

D. Gómez Rodríguez, Antonio

- Ingeniero de soluciones Cloud en Oracle
- Director de Proyectos en Sopra Group
- Director de Proyectos en Everis
- Jefe de Proyectos en la empresa pública de Gestión de Programas Culturales de Consejería de Cultura de Andalucía
- Profesor de la asignatura Blockchain Empresariales – Oracle" en el Curso de Extensión Universitaria en Tecnologías Blockchain de la Universidad de Málaga
- Licenciado en Ingeniería Superior de Telecomunicación por la Universidad Politécnica de Cataluña
- Postgrado en Tecnologías y Sistemas de Información por el Instituto Catalán de Tecnología.
- E-Business Master en la Escuela de Negocios La Salle

D. Bernal de la Varga, Yeray

- Arquitecto de Soluciones Big Data en Orange Bank
- Arquitecto Big Data en Bankia
- Ingeniero Big Data en Hewlett-Packard
- Profesor adjunto en el Master de Big Data por la Universidad de Deusto
- Licenciado en Informática por la Universidad Politécnica de Madrid
- Experto en Big Data por la U-TAD

Dña. Rodríguez Camacho, Cristina

- Consultora de Apis y desarrolladora de microservicios en Inetum
- Graduada en Ingeniería de la salud, con mención en Ingeniería Biomédica por la Universidad de Málaga
- Máster en *Blockchain* y Big Data por la Universidad Complutense de Madrid
- Experta en Devops & Cloud por la UNIR

D. Torres Palomino, Sergio

- Ingeniero Big Data y técnico ITIL en IECISA
- Ingeniero Big Data en GOLIVE SERVICES
- Desarrollador Blockchain en Blocknitive
- Arquitecto Blockchain en Signeblock y TELEFÓNICA
- Docente de Hyperledger en los Másteres en Blockchain de la Universidad Europea, la Universidad de Valladolid, Three Points y la Universidad Complutense
- Graduado en Ingeniería Informática por la Universidad San Pablo CEU
- Máster en Arquitectura Big Data por CIFF Universidad de Alcalá
- Máster en Big Data y Business Analytic por KSCHOOL, Madrid

D. Rodríguez García, Darío

- Arquitecto de Software en NEA F3 MASTER
- Profesor en el Máster Universitario en Ingeniería Web en la Universidad de Oviedo
- Desarrollador Full-Stack en NEA F3 MASTER
- Instructor de cursos en la plataforma de e-learning Udemy
- Graduado en Ingeniería Informática del Software por la Universidad de Oviedo
- Máster Universitario en Ingeniería Web por la Universidad de Oviedo

Dr. Moguel Márquez, Miguel

- Ingeniero Informático
- Doctor en Tecnologías Informáticas por la Universidad de Extremadura
- Máster en Ingeniería Informática por la Universidad de Extremadura
- Máster Universitario en Formación de Profesores de Educación Secundaria
- Grado en Ingeniería Informática
- Docente en el ámbito de la Ingeniería Web, diseño y desarrollo de aplicaciones en la Web, Arquitecturas Software y nuevas tendencias tecnológicas

Dr. García Sanz-Calcedo, Justo

- Director de Ingeniería y Mantenimiento en Servicio Extremeño de Salud
- Doctor Ingeniero Industrial por la Universidad de Extremadura
- Ingeniería Industrial por la Universidad de Extremadura
- Experto en Habilidades para la Dirección de Equipos y Formador de Formadores
- Profesor Titular de la Universidad de Extremadura
- Programa de Alta Dirección en Instituciones Sanitarias por IESE Business School





Dr. Sánchez-Barroso Moreno, Gonzalo

- Consultor de proyectos de Investigación Industrial y Desarrollo Experimental
- Doctor Ingeniero Industrial por la Universidad de Extremadura
- Máster en Ingeniería Industrial y Grado en Ingeniería Mecánica (Rama Industrial) por la Universidad de Extremadura
- Especialización en Dirección de Proyectos de Innovación
- Certified Project Management Associate (Level D) by International Project Management Association (IPMA)
- Scrum Foundation Professional Certificate (SFPC)

Dr. González Domínguez, Jaime

- Consultor de proyectos de Investigación Industrial y Desarrollo Experimental
- Doctorado en Modelización y Experimentación en Ciencia y Tecnología
- Ingeniero Industrial Ingeniero Mecánico (Rama Industrial) por la Universidad de Extremadura
- Especialización en Dirección de Proyectos de Innovación
- Certified Project Management Associate (Level D) by International Project Management Association (IPMA)
- Scrum Foundation Professional Certificate (SFPC)

05

Estructura y contenido

El temario se ha diseñado en base a los exhaustivos requisitos del equipo docente que conforma esta titulación. De esta forma, se ha establecido un plan de estudio conformado por diez módulos que ofrecen una amplia y detallada visión sobre los entornos *Cloud*, las diferentes herramientas existentes y sus posibilidades en un campo emergente. El profesional de la informática que curse esta enseñanza será capaz de programar con aplicaciones *Cloud Natives*, diseñar e implementar una red segura o realizar la programación *Cloud* en tiempo real. Todo ello, con el respaldo de un amplio contenido multimedia rico en vídeos en detalle, lecturas adicionales y ejemplos prácticos reales que complementan esta enseñanza.





TECH te ofrece un plan de estudios con contenido de calidad y con un enfoque actual e innovador en los entornos Cloud"

Módulo 1. Programación *Cloud*. Servicios en Azure, Aws y Google *Cloud*

1.1. *Cloud*. Servicios y Tecnologías *Cloud*
- 1.1.1. Servicios y Tecnologías *Cloud*
- 1.1.2. Terminología *Cloud*
- 1.1.3. Proveedores *Cloud* de referencia

1.2. *Cloud Computing*
- 1.2.1. *Cloud Computing*
- 1.2.2. Ecosistema del *Cloud Computing*
- 1.2.3. Tipología *Cloud Computing*

1.3. Modelos de Servicio en *Cloud*
- 1.3.1. IaaS. Infraestructura como Servicio
- 1.3.2. SaaS. Software como Servicio
- 1.3.3. PaaS. Plataforma como Servicio

1.4. Tecnologías *Cloud Computing*
- 1.4.1. Sistema de Virtualización
- 1.4.2. Service-Oriented Architecture (SOA)
- 1.4.3. Computación GRID

1.5. Arquitectura *Cloud Computing*
- 1.5.1. Arquitectura *Cloud Computing*
- 1.5.2. Tipologías de red en Cloud Computing
- 1.5.3. Seguridad en *Cloud Computing*

1.6. *Public Cloud*
- 1.6.1. *Public Cloud*
- 1.6.2. Arquitectura y costes de *Public Cloud*
- 1.6.3. *Public Cloud*. Tipología

1.7. *Private Cloud*
- 1.7.1. *Private Cloud*
- 1.7.2. Arquitectura y Costes
- 1.7.3. *Private Cloud*. Tipología

1.8. *Hybrid Cloud*
- 1.8.1. *Hybrid Cloud*
- 1.8.2. Arquitectura y costes
- 1.8.3. *Hybrid Cloud*. Tipología

1.9. Proveedores *Cloud*
- 1.9.1. Amazon Web Services
- 1.9.2. Azure
- 1.9.3. Google

1.10. Seguridad en *Cloud*
- 1.10.1. Seguridad en Infraestructura
- 1.10.2. Seguridad en el Sistema Operativo y Redes
- 1.10.3. Mitigación de riesgos en *Cloud*

Módulo 2. Programación de Arquitecturas en *Cloud Computing*

2.1. Arquitectura *Cloud* para una red universitaria. Selección del Proveedor *Cloud*. Ejemplo Práctico
- 2.1.1. Planteamiento de Arquitectura *Cloud* para una red universitaria según proveedor *Cloud*
- 2.1.2. Componentes de Arquitectura *Cloud*
- 2.1.3. Análisis de las soluciones *Cloud* según Arquitectura propuesta

2.2. Estimación económica del proyecto de creación de una red universitaria. Financiación
- 2.2.1. Selección del proveedor *Cloud*
- 2.2.2. Estimación económica en base a los componentes
- 2.2.3. Financiación del proyecto

2.3. Estimación de Recursos Humanos del Proyecto. Composición de un equipo software
- 2.3.1. Composición del equipo de desarrollo software
- 2.3.2. Roles en un equipo de desarrollo. Tipología
- 2.3.3. Evaluación de la estimación económica del proyecto

2.4. Cronograma de ejecución y documentación del proyecto
- 2.4.1. Cronograma agile del proyecto
- 2.4.2. Documentación para la viabilidad del proyecto
- 2.4.3. Documentación a aportar para la ejecución del proyecto

2.5. Implicaciones legales de un proyecto
- 2.5.1. Implicaciones legales de un proyecto
- 2.5.2. Política de Protección de Datos
  - 2.5.2.1. GDPR. Reglamento General de Protección de Datos
- 2.5.3. Responsabilidad de la empresa integradora

2.6. Diseño y creación de una red *Blockchain* en *Cloud* para la arquitectura propuesta
- 2.6.1. *Blockchain* – Hyperledger Fabric
- 2.6.2. Hyperledger Fabric Basics
- 2.6.3. Diseño de una red de Hyperledger Fabric universitaria internacional

2.7. Planteamiento de ampliación de la arquitectura propuesta
- 2.7.1. Creación de la arquitectura propuesta con *Blockchain*
- 2.7.2. Ampliación de la arquitectura propuesta
- 2.7.3. Configuración de una arquitectura en alta disponibilidad

2.8. Administración de la arquitectura *Cloud* propuesta
- 2.8.1. Suma de un nuevo participante a la arquitectura propuesta inicial
- 2.8.2. Administración de la arquitectura *Cloud*
- 2.8.3. Gestión de la lógica del proyecto – *Smart Contracts*

2.9. Administración y gestión de los componentes específicos en la arquitectura *Cloud* propuesta
- 2.9.1. Gestión de los certificados de una red
- 2.9.2. Gestión de la seguridad de diversos componentes: CouchDB
- 2.9.3. Gestión de los nodos de la red *blockchain*

2.10. Modificación de una instalación básica inicial en la creación de la red *blockchain*
- 2.10.1. Suma de un nodo a la red *blockchain*
- 2.10.2. Suma de persistencia de datos extra
- 2.10.3. Gestión de *Smart Contracts*
- 2.10.4. Suma de una nueva universidad a la red existente

Módulo 3. Storage en *Cloud* Azure

3.1. Instalación MV en Azure
- 3.1.1. Comandos de creación
- 3.1.2. Comandos de visualización
- 3.1.3. Comandos de modificación

3.2. Blobs en Azure
- 3.2.1. Tipos de Blob
- 3.2.2. Contenedor
- 3.2.3. Azcopy
- 3.2.4. Supresión reversible de blobs

3.3. Disco Administrado y Almacenamiento en Azure
- 3.3.1. Disco administrado
- 3.3.2. Seguridad
- 3.3.3. Almacenamiento en frío
- 3.3.4. Replicación
  - 3.3.4.1. Redundancia local
  - 3.3.4.2. Redundancia en una zona
  - 3.3.4.3. Georredundante

3.4. Tablas, Colas, Archivos en Azure
- 3.4.1. Tablas
- 3.4.2. Colas
- 3.4.3. Archivos

3.5. Encriptación y Seguridad en Azure
- 3.5.1. *Storage Service Encryption* (SSE)
- 3.5.2. Claves de acceso
  - 3.5.2.1. Firma de acceso compartido
  - 3.5.2.2. Directivas de acceso a nivel de contenedores
  - 3.5.2.3. Firma de acceso a nivel de blob
- 3.5.3. Autenticación Azure AD

3.6. Red Virtual en Azure
- 3.6.1. Subred y Emparejamiento
- 3.6.2. Vnet to Vnet
- 3.6.3. Enlace privado
- 3.6.4. Alta disponibilidad

3.7. Tipos de conexiones en Azure
- 3.7.1. Azure *Application Gateway*
- 3.7.2. VPN de sitio a sitio
- 3.7.3. VPN punto a sitio
- 3.7.4. *ExpressRoute*

3.8. Recursos en Azure
- 3.8.1. Bloqueo de recursos
- 3.8.2. Movimiento de recursos
- 3.8.3. Eliminación de recursos

3.9. *Backup* en Azure
- 3.9.1. *Recovery Services*
- 3.9.2. Agente Azure *Backup*
- 3.9.3. Azure *Backup Server*

3.10. Desarrollo de Soluciones
- 3.10.1. Compresión, deduplicación, replicación
- 3.10.2. *Recovery Services*
- 3.10.3. *Disaster Recovery Plan*

Módulo 4. Entornos *Cloud*. Seguridad

4.1. Entornos *Cloud*. Seguridad
- 4.1.1. Entornos *Cloud*, seguridad
  - 4.1.1.1 Seguridad en *Cloud*
  - 4.1.1.2. Postura de seguridad

4.2. Modelo de Gestión de seguridad compartida en *Cloud*
- 4.2.1. Elementos de seguridad gestionados por proveedor
- 4.2.2. Elementos gestionados por cliente
- 4.2.3. Estrategia para seguridad

4.3. Mecanismos de prevención en *Cloud*
- 4.3.1. Sistemas de gestión de autenticación
- 4.3.2. Sistema de gestión de autorización. Políticas de acceso
- 4.3.3. Sistemas de gestión de claves

4.4. Seguridad de los datos en infraestructura *Cloud*
- 4.4.1. Securización de los sistemas de almacenamiento:
  - 4.4.1.1. *Block*
  - 4.4.1.2. *Object storage*
  - 4.4.1.3. *Ffile systems*
- 4.4.2. Protección de los sistemas de base de datos
- 4.4.3. Securización de datos en tránsito

4.5. Protección de Infraestructura *Cloud*
- 4.5.1. Diseño e implementación de red segura
- 4.5.2. Seguridad en recursos de computación
- 4.5.3. Herramientas y recursos para protección de infraestructura



4.6. Riesgos y vulnerabilidades en aplicaciones
- 4.6.1. Riesgos en desarrollo de aplicaciones
- 4.6.2. Riesgos de seguridad críticos
- 4.6.3. Vulnerabilidades en el desarrollo de software

4.7. Defensas en aplicaciones frente a ataques
- 4.7.1. Diseño en el desarrollo de aplicaciones
- 4.7.2. Securización a través de la verificación y testeo
- 4.7.3. Práctica de programación segura

4.8. Seguridad en Entornos DevOps
- 4.8.1. Seguridad en entornos virtualizados y con *containers*
- 4.8.2. Seguridad en Desarrollo y Operaciones (DevSecOps)
- 4.8.3. Mejores prácticas en seguridad en entornos productivos con *containers*

4.9. Seguridad en *Clouds* Públicos
- 4.9.1. AWS
- 4.9.2. Azure
- 4.9.3. Oracle *Cloud*

4.10. Normativa de Seguridad, gobernanza y cumplimiento
- 4.10.1. Cumplimiento de normativas de seguridad
- 4.10.2. Gestión de riesgos
- 4.10.3. Proceso en las organizaciones

Módulo 5. Orquestación contenedores: Kubernetes y Docker

5.1. Base de arquitecturas de aplicaciones
- 5.1.1. Modelos de aplicaciones actuales
- 5.1.2. Plataformas de ejecución de aplicaciones
- 5.1.3. Tecnologías de contenedores

5.2. Arquitectura de Docker
- 5.2.1. Arquitectura de Docker
- 5.2.2. Instalación arquitectura Docker
- 5.2.3. Comandos. Proyecto local

5.3. Arquitectura Docker. Gestión del almacenamiento
- 5.3.1. Manejo de imágenes y registro
- 5.3.2. Redes en Docker
- 5.3.3. Gestión del almacenamiento

5.4. Arquitectura Docker avanzado
- 5.4.1. Docker Compose
- 5.4.2. Docker en la organización
- 5.4.3. Ejemplo de adopción de Docker

5.5. Arquitectura Kubernetes
- 5.5.1. Arquitectura Kubernetes
- 5.5.2. Elementos de despliegue en Kubernetes
- 5.5.3. Distribuciones y soluciones gestionadas
- 5.5.4. Instalación y entorno

5.6. Arquitecturas Kubernetes: Desarrollo con Kubernetes
- 5.6.1. Herramientas para el desarrollo en K8s
- 5.6.2. Modo imperativo vs declarativo
- 5.6.3. Despliegue y exposición de aplicaciones

5.7. Kubernetes en entornos empresariales
- 5.7.1. Persistencia de datos
- 5.7.2. Alta disponibilidad, escalado y red
- 5.7.3. Seguridad en Kubernetes
- 5.7.4. Gestión y monitorización de Kubernetes

5.8. Distribuciones de K8s
- 5.8.1. Comparativa de entornos de despliegue
- 5.8.2. Despliegue en GKE, AKS, EKS o OKE
- 5.8.3. Despliegue *on premise*

5.9. Rancher y Openshift
- 5.9.1. Rancher
- 5.9.2. Openshift
- 5.9.3. Openshift: configuración y despliegue de aplicaciones

5.10. Arquitecturas Kubernetes y Containers. Actualizaciones
- 5.10.1. *Open Application Model*
- 5.10.2. Herramientas para gestión de despliegue en entornos Kubernetes
- 5.10.3. Referencias a otros proyectos y tendencias

Módulo 6. Programación de aplicaciones *Cloud* Nativas

6.1. Tecnologías *Cloud-Native*
- 6.1.1. Tecnologías *Cloud-Native*
- 6.1.2. *Cloud Native Computing Foundation*
- 6.1.3. Herramientas para desarrollo *Cloud-Native*

6.2. Arquitectura de Aplicaciones *Cloud-Native*
- 6.2.1. Diseño de aplicaciones *Cloud-Native*
- 6.2.2. Componentes de Arquitectura *Cloud-Native*
- 6.2.3. Modernización de Aplicaciones *Legacy*

6.3. *Containerization*
- 6.3.1. Desarrollo con orientación a *Containers*
- 6.3.2. Desarrollo con Microservicios
- 6.3.3. Herramientas para el trabajo en equipo

6.4. DevOps y la integración y despliegues continuos
- 6.4.1. Integración y despliegues continuos: CI/CD
- 6.4.2. Ecosistema de herramientas para CI/CD
- 6.4.3. Creación de un entorno de CI/CD

6.5. Observabilidad y análisis de la plataforma
- 6.5.1. Observabilidad de Aplicaciones *Cloud-Native*
- 6.5.2. Herramientas para Monitorización, *Logging* y trazabilidad
- 6.5.3. Puesta en marcha de un entorno de observabilidad y análisis

6.6. Gestión de datos en aplicaciones *cloud-native*
- 6.6.1. Base de Datos en *Cloud-Native*
- 6.6.2. Patrones en la Gestión de Datos
- 6.6.3. Tecnologías para implementar los Patrones en Gestión de Datos

6.7. Comunicaciones en las Aplicaciones *Cloud-Native*
- 6.7.1. Comunicaciones síncronas y asíncronas
- 6.7.2. Tecnologías para patrones de comunicaciones síncronos
- 6.7.3. Tecnologías para patrones de comunicaciones asíncronos

6.8. Resiliencia, seguridad y rendimiento en las aplicaciones *Cloud-Native*
- 6.8.1. Resiliencia de las aplicaciones
- 6.8.2. Desarrollo seguro en aplicaciones *Cloud-Native*
- 6.8.3. Rendimiento y Escalabilidad de las Aplicaciones

6.9. Serverless
- 6.9.1. Serverless en *Cloud-Native*
- 6.9.2. Plataformas de *Serverless*
- 6.9.3. Casos de uso para desarrollo *Serverless*

6.10. Plataformas de Despliegue
- 6.10.1. Entornos para desarrollos *Cloud-Native*
- 6.10.2. Plataformas de orquestación. Comparativa
- 6.10.3. Automatización de Infraestructura

Módulo 7. Programación *Cloud*. Data *Governance*

7.1. Gestión de datos
- 7.1.1. Gestión de datos
- 7.1.2. Ética en el manejo de datos

7.2. Data *Governance*
- 7.2.1. Clasificación. Control de acceso
- 7.2.2. Regulación sobre el Tratamiento de Datos
- 7.2.3. Data *Governance*. Valor

7.3. Gobierno de Datos. Herramientas
- 7.3.1. Linaje
- 7.3.2. Metadatos
- 7.3.3. Catálogo de datos. *Business Glossary*

7.4. Usuarios y procesos en el gobierno de datos
- 7.4.1. Usuarios
  - 7.4.1.1. Roles y responsabilidades
- 7.4.2. Procesos
  - 7.4.2.1. Enriquecimiento de datos

7.5. Ciclo de vida de los datos en la empresa
- 7.5.1. Creación de los datos
- 7.5.2. Procesamiento de datos
- 7.5.3. Almacenamiento de datos
- 7.5.4. Uso de los datos
- 7.5.5. Destrucción de los datos

7.6. Calidad del dato
- 7.6.1. La calidad en el gobierno del dato
- 7.6.2. Calidad del dato en analítica
- 7.6.3. Técnicas de calidad del dato

7.7. Gobierno del dato en tránsito
- 7.7.1. Gobierno del dato en tránsito
  - 7.7.1.1. Linaje
- 7.7.2. La cuarta dimensión

7.8. Protección de datos
- 7.8.1. Niveles de acceso
- 7.8.2. Clasificación
- 7.8.3. *Compliance*. Normativa

7.9. Monitorización y Medida del Gobierno del dato
- 7.9.1. Monitorización y medida del Gobierno del dato
- 7.9.2. Monitorización del linaje
- 7.9.3. Monitorización de la calidad del dato

7.10. Herramientas para el gobierno del dato
- 7.10.1. Talend
- 7.10.2. Collibra
- 7.10.3. Informática

Módulo 8. Programación Cloud en Tiempo Real. *Streaming*

8.1. Procesamiento y estructuración de la información en *streaming*
- 8.1.1. Proceso de recolección, estructuración, procesado, análisis e interpretación de los datos
- 8.1.2. Técnicas de procesamiento de datos en *streaming*
- 8.1.3. Procesamiento en *streaming*
- 8.1.4. Casos de uso del procesamiento en *streaming*

8.2. Estadística para la comprensión del flujo del dato *streaming*
- 8.2.1. Estadística descriptiva
- 8.2.2. Cálculo de probabilidades
- 8.2.3. Inferencia

8.3. Programación con Python
- 8.3.1. Tipología, condicionales, funciones y bucles
- 8.3.2. Numpy, matplotlib, dataframes, ficheros csv y formatos json
- 8.3.3. Secuencias: listas, bucles, ficheros y diccionarios
- 8.3.4. Mutabilidad, excepciones y funciones de orden superior

8.4. Programación con R
- 8.4.1. Programación con R
- 8.4.2. Vectores y factores
- 8.4.3. Matrices y *arrays*
- 8.4.4. Listas y data *frame*
- 8.4.5. Funciones

8.5. Base de datos SQL para el Procesamiento de Datos en *streaming*
- 8.5.1. Base de datos SQL
- 8.5.2. Modelo entidad - relación
- 8.5.3. Modelo relacional
- 8.5.4. SQL

8.6. Base de datos NO SQL para el Procesamiento de Datos en *streaming*
- 8.6.1. Base de datos NO SQL
- 8.6.2. MongoDB
- 8.6.3. Arquitectura MongoDB
- 8.6.4. Operaciones CRUD
- 8.6.5. *Find*, proyecciones, indexes aggregation y cursores
- 8.6.6. Modelo de datos

8.7. Minería de datos y modelización predictiva
- 8.7.1. Análisis multivariante
- 8.7.2. Técnicas de reducción de la dimensión
- 8.7.3. Análisis clúster
- 8.7.4. Series

8.8. *Maching learning* para Procesamiento de Datos en *streaming*
- 8.8.1. *Maching learning* y modelización predictiva avanzada
- 8.8.2. Redes Neuronales
- 8.8.3. *Deep Learning*
- 8.8.4. *Bagging* y *Random Forest*
- 8.8.5. *Gradient Bosting*
- 8.8.6. SVM
- 8.8.7. Métodos de ensamblado

8.9. Tecnologías en el procesamiento de datos en *streaming*
- 8.9.1. Spark *Streaming*
- 8.9.2. Kafka *Streams*
- 8.9.3. Flink *Streaming*

8.10. Apache Spark *Streaming*
- 8.10.1. Apache Spark *Streaming*
- 8.10.2. Componentes de Spark
- 8.10.3. Arquitectura de Spark
- 8.10.4. RDD
- 8.10.5. SPARK SQL
- 8.10.6. *Jobs, stages* y *task*





Módulo 9. Integración *Cloud* con Servicios Web. Tecnologías y Protocolos

9.1. Estándares y protocolos de la Web
- 9.1.1. Web y Web 2.0
- 9.1.2. Arquitectura cliente-servidor
- 9.1.3. Protocolos y estándares de comunicación

9.2. Servicios Web
- 9.2.1. Los servicios web
- 9.2.2. Capas y mecanismos de comunicación
- 9.2.3. Arquitecturas de servicios

9.3. Arquitecturas Orientadas a Servicios
- 9.3.1. *Service Oriented Architecture* (SOA)
- 9.3.2. Diseño de servicios web
- 9.3.3. SOAP y REST

9.4. SOAP. Service Oriented Arquitecture
- 9.4.1. Estructura y paso de mensajes
- 9.4.2. *Web Service Description Language* (WSDL)
- 9.4.3. Implementación de clientes y servidores SOAP

9.5. Arquitecturas REST
- 9.5.1. Las arquitecturas REST y Servicios Web RESTful
- 9.5.2. Verbos HTTP: semántica y propósitos
- 9.5.3. *Swagger*
- 9.5.4. Implementación de clientes y servidores REST

9.6. Arquitecturas basadas en microservicios
- 9.6.1. Planteamiento monolítico de arquitectura vs. uso microservicios
- 9.6.2. Las Arquitecturas basadas en microservicios
- 9.6.3. Flujos de comunicación con el uso de microservicios

9.7. Invocación de APIs desde el lado cliente
- 9.7.1. Tipologías de clientes Web
- 9.7.2. Herramientas de desarrollo para el tratamiento de servicios Web
- 9.7.3. Recursos de Origen Cruzado (CORS)

9.8. Seguridad en la invocación a APIs
- 9.8.1. Seguridad en los Servicios Web
- 9.8.2. Autenticación y autorización
- 9.8.3. Métodos de autenticación en base al grado de seguridad

9.9. Integración de aplicaciones con proveedores *Cloud*
- 9.9.1. Proveedores de *Cloud Computing*
- 9.9.2. Servicios de las plataformas
- 9.9.3. Servicios orientados a la implementación/consumo de Servicios Web

9.10. Implementación de *Bots* y Asistentes
- 9.10.1. Uso de *Bots*
- 9.10.2. Uso del Servicio Web en *Bots*
- 9.10.3. Implementación de *Chatbots* y asistentes web

Módulo 10. Programación *Cloud*. Gestión de proyectos y verificación del producto

10.1. Metodologías Cascada
- 10.1.1. Clasificación de metodologías
- 10.1.2. Modelo en cascada. *Waterfall*
- 10.1.3. *Strong and weakness*
- 10.1.4. Comparativa de modelos. *Waterfall* vs. *Agile*

10.2. Metodología *Agile*
- 10.2.1. Metodología *Agile*
- 10.2.2. El manifiesto *Agile*
- 10.2.3. Uso de *Agile*

10.3. Metodología Scrum
- 10.3.1. Metodología Scrum
  - 10.3.1.1. Uso de Scrum
- 10.3.2. Eventos de Scrum
- 10.3.3. Artefactos de Scrum
- 10.3.4. Guía de Scrum

10.4. *Agile Inception Desk*
- 10.4.1. *Agile Inception Desk*
- 10.4.2. Fases en *Inception Desk*

10.5. Técnica *Impact Mapping*
- 10.5.1. *Impact Mapping*
- 10.5.2. Uso de *Impact Mappig*
- 10.5.3. Estructura *Impact Mapping*

10.6. Historias de usuario
- 10.6.1. Historias de usuario
- 10.6.2. Redacción de historias de usuario
- 10.6.3. Jerarquía de historias de usuario
- 10.6.4. *Use Story Mapping*

10.7. Test Qa Manual
- 10.7.1. Testing manual
- 10.7.2. Validación y Verificación. Diferencias
- 10.7.3. Pruebas manuales. Tipología
- 10.7.4. UAT. *User Acceptance Testing*
- 10.7.5. UAT y pruebas Alfa & beta
- 10.7.6. Calidad del software

10.8. Pruebas Automáticas
- 10.8.1. Pruebas Automáticas
- 10.8.2. Pruebas Manuales vs. Automáticas
- 10.8.3. El impacto del test automático
- 10.8.4. El Resultado de aplicar automatización
- 10.8.5. La rueda de la calidad

10.9. Pruebas Funcionales y No Funcionales
- 10.9.1. Pruebas funcionales y no funcionales
- 10.9.2. Pruebas funcionales
  - 10.9.2.1. Pruebas unitarias
  - 10.9.2.2. Pruebas de integración
  - 10.9.2.3. Pruebas de regresión
  - 10.9.2.4. Pruebas de *smoke* test
  - 10.9.2.5. Pruebas de mono
  - 10.9.2.6. Pruebas de sanidad



10.9.3. Pruebas No funcionales
10.9.3.1. Pruebas de carga
10.9.3.2. Pruebas de rendimiento
10.9.3.3. Pruebas de seguridad
10.9.3.4. Pruebas de configuración
10.9.3.5. Pruebas de estrés
10.10. Métodos y Herramientas de Verificación
10.10.1. Mapa de calor
10.10.2. *Eye tracking*
10.10.3. Mapas de *Scroll*
10.10.4. Mapas de movimiento
10.10.5. Mapas de confeti
10.10.6. Test A/B
10.10.7. Método *Blue & Green Deployment*
10.10.8. Método *Canary Release*
10.10.9. Selección de las herramientas
10.10.10. Herramientas analíticas

Sé un auténtico profesional. Reduce los riesgos en Cloud y garantiza seguridad a las empresas para las que trabajes"

07

Titulación

El Máster Título Propio en Programación Cloud garantiza, además de la capacitación más rigurosa y actualizada, el acceso a un título de Máster Propio expedido por TECH Universidad Tecnológica.





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

Este **Máster Título Propio en Programación Cloud** contiene el programa más completo y actualizado del mercado.

Tras la superación de la evaluación, el alumno recibirá por correo postal* con acuse de recibo su correspondiente título de **Máster Propio** emitido por **TECH Universidad Tecnológica.**

El título expedido por **TECH Universidad Tecnológica** expresará la calificación que haya obtenido en el Máster Título Propio, y reunirá los requisitos comúnmente exigidos por las bolsas de trabajo, oposiciones y comités evaluadores de carreras profesionales.

Título: **Máster Título Propio en Programación Cloud**

Nº Horas Oficiales: **1.500 h.**



tech universidad tecnológica

Otorga la presente

CONSTANCIA

a

C. _____, con documento de identificación nº_____

Por haber superado con éxito y acreditado el programa de

MÁSTER TÍTULO PROPIO

en

Programación Cloud

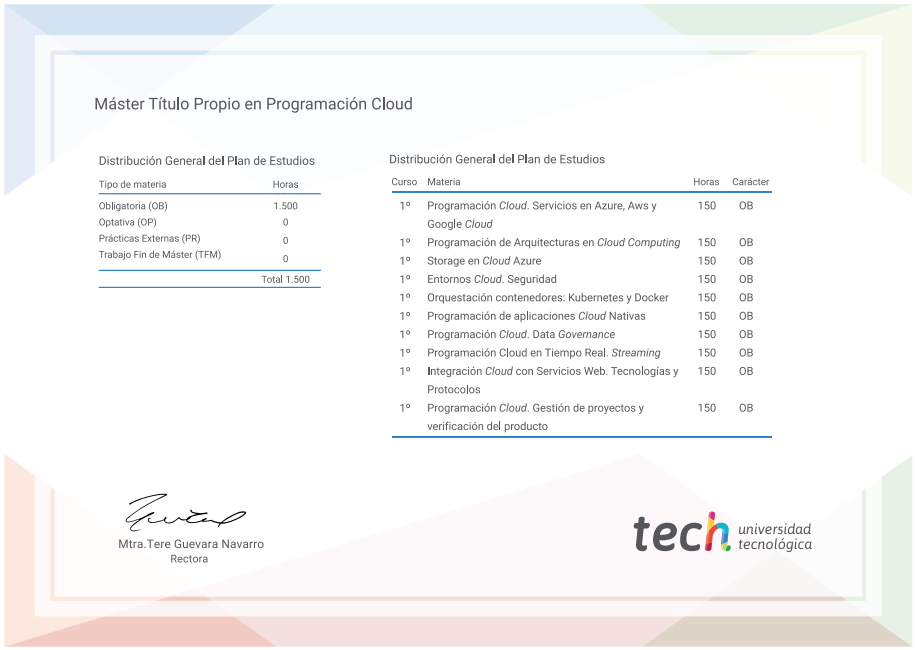
Se trata de un título propio de esta Universidad con una duración de 1.500 horas, con fecha de inicio dd/mm/aaaa y fecha de finalización dd/mm/aaaa.

TECH es una Institución Particular de Educación Superior reconocida por la Secretaría de Educación Pública a partir del 28 de junio de 2018.

A 17 de junio de 2020

Mtra.Tere Guevara Navarro
Rectora

Este título propio se deberá acompañar siempre del título universitario habilitante expedido por la autoridad competente para ejercer profesionalmente en cada país. código único TECH: AFWOR23S techtitute.com/titulos



Máster Título Propio en Programación Cloud

Distribución General del Plan de Estudios

Tipo de materia	Horas
Obligatoria (OB)	1.500
Optativa (OP)	0
Prácticas Externas (PR)	0
Trabajo Fin de Máster (TFM)	0
	Total 1.500

Distribución General del Plan de Estudios

Curso	Materia	Horas	Carácter
1º	Programación *Cloud*. Servicios en Azure, Aws y Google *Cloud*	150	OB
1º	Programación de Arquitecturas en *Cloud Computing*	150	OB
1º	Storage en *Cloud* Azure	150	OB
1º	Entornos *Cloud*. Seguridad	150	OB
1º	Orquestación contenedores: Kubernetes y Docker	150	OB
1º	Programación de aplicaciones *Cloud* Nativas	150	OB
1º	Programación *Cloud*. Data *Governance*	150	OB
1º	Programación Cloud en Tiempo Real. *Streaming*	150	OB
1º	Integración *Cloud* con Servicios Web. Tecnologías y Protocolos	150	OB
1º	Programación *Cloud*. Gestión de proyectos y verificación del producto	150	OB

Mtra.Tere Guevara Navarro
Rectora

tech universidad tecnológica

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

tech universidad tecnológica

Máster Título Propio
Programación Cloud

Modalidad: **Online**
Duración: **12 meses**
Titulación: **TECH Universidad Tecnológica**
Horas lectivas: **1.500 h.**

Máster Título Propio
Programación Cloud



tech universidad tecnológica