# Experto Universitario
## Ingeniería del Software



tech universidad tecnológica



tech universidad tecnológica

# Experto Universitario
## Ingeniería del Software

Modalidad: **Online**
Duración: **6 meses**
Titulación: **TECH - Universidad Tecnológica**
Horas lectivas: **600 h.**
Acceso web: www.techtitute.com/informatica/experto-universitario/experto-ingenieria-software

# Índice


01 Presentación — pág. 4

02 Objetivos — pág. 8

03 Estructura y contenido — pág. 12

04 Metodología — pág. 22

05 Titulación — pág. 30

01

# Presentación

Actualiza tus conocimientos en Ingeniería del Software con los mejores profesionales del sector. Con esta formación de alto nivel, aprenderás las últimas tendencias y novedades en la materia. Una oportunidad única de dar el impulso que tu carrera necesita.





*Este Experto te permitirá actualizar tus conocimientos en Ingeniería del Software de un modo práctico, 100% online, sin renunciar al máximo rigor académico"*

Este programa está dirigido a aquellas personas interesadas en alcanzar un nivel de conocimiento superior en Ingeniería del Software. El principal objetivo es formar al alumno para que aplique en el mundo real los conocimientos adquiridos en este Experto, en un entorno de trabajo que reproduzca las condiciones que se puede encontrar en su futuro, de manera rigurosa y realista.

Este Experto te preparará para el ejercicio profesional de la ingeniería informática, gracias a una formación transversal y versátil adaptada a las nuevas tecnologías e innovaciones en este campo. Obtendrás amplios conocimientos en Ingeniería del Software, de la mano de profesionales en el sector.

Aprovecha la oportunidad y cursa esta formación en un formato 100% online, sin tener que renunciar a tus obligaciones. Actualiza tus conocimientos y consigue tu título de Experto para seguir creciendo personal y profesionalmente.

*Aprende las últimas técnicas y estrategias con este programa y alcanza el éxito como ingeniero informático"*

Este **Experto Universitario en Ingeniería del Software** contiene el programa científico más completo y actualizado del mercado. Las características más destacadas del curso son:

- Desarrollo de 100 escenarios simulados presentados por expertos en Ingeniería del Software.
- Sus contenidos gráficos, esquemáticos y eminentemente prácticos con los que están concebidos, recogen una información científica y práctica sobre el Ingeniería del Software.
- Novedades sobre los últimos avances en el Ingeniería del Software.
- Contiene ejercicios prácticos donde realizar el proceso de autoevaluación para mejorar el aprendizaje.
- Sistema interactivo de aprendizaje basado en el método del caso y su aplicación a la práctica real.
- Todo esto se complementará con lecciones teóricas, preguntas al experto, foros de discusión de temas controvertidos y trabajos de reflexión individual.
- Disponibilidad de los contenidos desde cualquier dispositivo fijo o portátil con conexión a internet.

*Este programa te permitirá potenciar tus capacidades y actualizar tus conocimientos en Ingeniería del Software”*

*Aprovecha la última tecnología educativa para ponerte al día en Ingeniería del Software sin moverte de casa.*

*Conoce las últimas técnicas en Ingeniería del Software de la mano de expertos en la materia.*

Incluye en su cuadro docente profesionales pertenecientes al ámbito de Ingeniería Informática, que vierten en esta formación la experiencia de su trabajo, además de reconocidos especialistas pertenecientes a sociedades de referencia y universidades de prestigio.

Gracias a su contenido multimedia elaborado con la última tecnología educativa, permitirán al profesional un aprendizaje situado y contextual, es decir, un entorno simulado que proporcionará un aprendizaje inmersivo programado para entrenarse ante situaciones reales.

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



02

# Objetivos

El objetivo de esta formación es ofrecer a los profesionales de Informática, los conocimientos y habilidades necesarios para realizar su actividad utilizando los protocolos y técnicas más avanzados del momento. Mediante un planteamiento de trabajo totalmente adaptable al alumno, este Experto te llevará progresivamente a adquirir las competencias que te impulsarán hacia un nivel profesional superior.





*Consigue el nivel de conocimiento que deseas y domina los conceptos fundamentales en Ingeniería del Software con esta formación de alto nivel"*

## Objetivos generales

- Formar científica y tecnológicamente, así como preparar para el ejercicio profesional de la ingeniería informática, todo ello con una formación transversal y versátil adaptada a las nuevas tecnologías e innovaciones en este campo.
- Obtener amplios conocimientos en el campo de la computación, la estructura de computadoras y la ingeniería del software, todo ello incluyendo la base matemática, estadística y física imprescindible en una ingeniería.

*Alcanza el éxito profesional como Ingeniero Informático con este programa intensivo, elaborado por profesionales con amplia experiencia en el sector"*





## Objetivos especificos por módulos

- Sentar las bases de la ingeniería del software y el modelado, aprendiendo los principales procesos y conceptos.
- Entender el proceso del software y los distintos modelos para su desarrollo incluyendo tecnologías ágiles.
- Comprender la ingeniería de requisitos, su desarrollo, elaboración, negociación y validación.
- Aprender el modelado de los requisitos y de los distintos elementos como escenarios, información, clases de análisis, flujo, comportamiento y patrones.
- Entender los conceptos y procesos del diseño de software, aprendiendo también sobre el diseño de la arquitectura y sobre el diseño a nivel de componentes y basado en patrones.
- Conocer las principales normas relativas a la calidad del software y a la administración de proyectos.
- Conocer en profundidad las distintas metodologías ágiles usadas en la ingeniería del software.
- Aprender desarrollar usando las técnicas de Scrum, programación extrema y de desarrollo de software basado en reutilización.
- Comprender los distintos patrones de arquitecturas de sistemas y de diseño de software, así como la arquitectura de las aplicaciones en la nube.
- Aprender a realizar pruebas al software, con metodologías como Test Driven Development, Acceptance Test Driven Development, Behavior Driven Development, BDD y Cucumber.
- Profundizar en la mejora del proceso de desarrollo de software y de calidad del software usando los estándares ISO/IEC.
- Introducir el concepto de DevOps y cuáles son sus principales prácticas.
- Comprender la importancia de la ingeniería de requisitos en el proceso de desarrollo de software.
- Profundizar en las fuentes de requisitos y las técnicas de elicitación de requisitos, ya que son parte esencial del proceso.
- Entender y aplicar la realización de prototipos como parte esencial del proceso de desarrollo.
- Aprender a realizar análisis de requisitos, así como a documentar adecuadamente los mismos.
- Entender los procesos de validación y negociación de requisitos, así como el modelado y gestión de requisitos.
- Adquirir los conocimientos necesarios para el manejo de sistemas críticos y la especificación formal de los requisitos
- Conocer el marco de referencia de la ingeniería del software y la norma ISO/IEC 12207.
- Aprender las características del proceso unificado de desarrollo de software y la planificación en el contexto del desarrollo de software ágil.
- Conocer los distintos estilos de diseño de software distribuido y arquitecturas software orientadas a servicios.
- Aprender los conceptos esenciales en el diseño de interfaces gráficas de usuario.
- Comprender las bases del desarrollo de aplicaciones web.
- Profundizar en las estrategias y técnicas de pruebas software, factores de calidad del software y distintas métricas utilizadas.

03

# Estructura y contenido

La estructura de los contenidos ha sido diseñada por un equipo de profesionales de Ingeniería Informática, conscientes de la relevancia de la actualidad de la formación para poder profundizar en esta área de conocimiento, con el fin de enriquecer humanísticamente al estudiante y elevarle el nivel de conocimiento en Ingeniería del Software mediante las últimas tecnologías educativas disponibles.





*Este Experto en Ingeniería del Software contiene el programa de aprendizaje más completo y actualizado del mercado"*

## **Módulo 1.** Ingeniería del Software

1.1. Introducción a la ingeniería del software y al modelado
- 1.1.1. La naturaleza del software
- 1.1.2. La naturaleza única de las webapps
- 1.1.3. Ingeniería del software
- 1.1.4. El proceso del software
- 1.1.5. La práctica de la ingeniería del software
- 1.1.6. Mitos del software
- 1.1.7. Cómo comienza todo
- 1.1.8. Conceptos orientados a objetos
- 1.1.9. Introducción a UML

1.2. El proceso del software
- 1.2.1. Un modelo general de proceso
- 1.2.2. Modelos de proceso prescriptivos
- 1.2.3. Modelos de proceso especializado
- 1.2.4. El proceso unificado
- 1.2.5. Modelos del proceso personal y del equipo
- 1.2.6. ¿Qué es la agilidad?
- 1.2.7. ¿Qué es un proceso ágil?
- 1.2.8. Scrum
- 1.2.9. Conjunto de herramientas para el proceso ágil

1.3. Principios que guían la práctica de la ingeniería del software
- 1.3.1. Principios que guían el proceso
- 1.3.2. Principios que guían la práctica
- 1.3.3. Principios de comunicación
- 1.3.4. Principios de planificación
- 1.3.5. Principios de modelado
- 1.3.6. Principios de construcción
- 1.3.7. Principios de despliegue

1.4. Comprensión de los requisitos
- 1.4.1. Ingeniería de requisitos
- 1.4.2. Establecer las bases
- 1.4.3. Indagación de los requisitos
- 1.4.4. Desarrollo de casos de uso
- 1.4.5. Elaboración del modelo de los requisitos
- 1.4.6. Negociación de los requisitos
- 1.4.7. Validación de los requisitos

1.5. Modelado de los requisitos: escenarios, información y clases de análisis
- 1.5.1. Análisis de los requisitos
- 1.5.2. Modelado basado en escenarios
- 1.5.3. Modelos UML que proporcionan el caso de uso
- 1.5.4. Conceptos de modelado de datos
- 1.5.5. Modelado basado en clases
- 1.5.6. Diagramas de clases

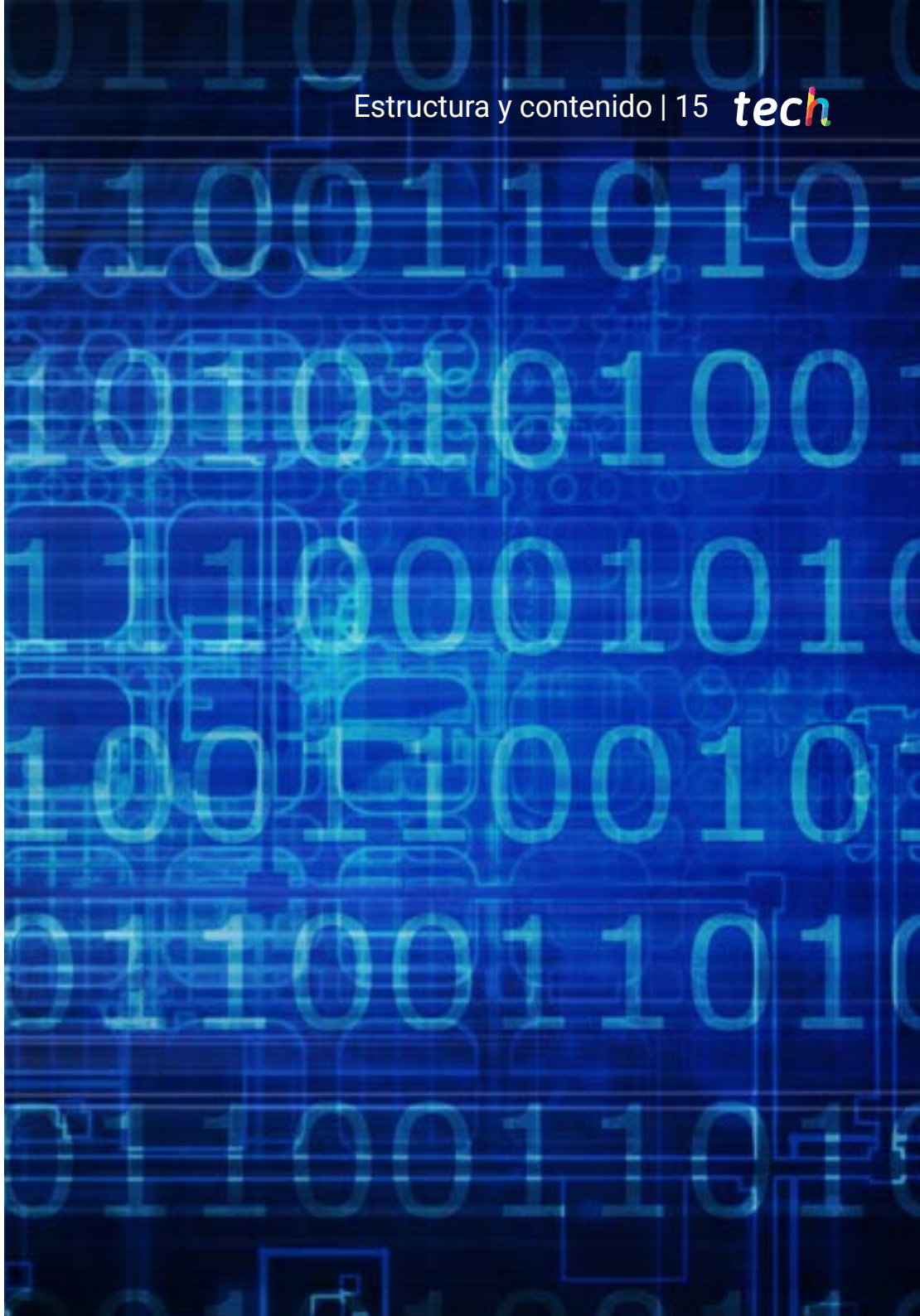
1.6. Modelado de los requisitos: flujo, comportamiento y patrones
- 1.6.1. Requisitos que modelan las estrategias
- 1.6.2. Modelado orientado al flujo
- 1.6.3. Diagramas de estado
- 1.6.4. Creación de un modelo de comportamiento
- 1.6.5. Diagramas de secuencia
- 1.6.6. Diagramas de comunicación
- 1.6.7. Patrones para el modelado de requisitos

1.7. Conceptos de diseño
- 1.7.1. Diseño en el contexto de la ingeniería del software
- 1.7.2. El proceso de diseño
- 1.7.3. Conceptos de diseño
- 1.7.4. Conceptos de diseño orientado a objetos
- 1.7.5. El modelo del diseño

1.8. Diseño de la arquitectura
- 1.8.1. Arquitectura del software
- 1.8.2. Géneros arquitectónicos
- 1.8.3. Estilos arquitectónicos
- 1.8.4. Diseño arquitectónico
- 1.8.5. Evolución de los diseños alternativos para la arquitectura
- 1.8.6. Mapeo de la arquitectura con el uso del flujo de datos

1.9. Diseño en el nivel de componentes y basado en patrones
- 1.9.1. ¿Qué es un componente?
- 1.9.2. Diseño de componentes basados en clase
- 1.9.3. Realización del diseño en el nivel de componentes
- 1.9.4. Diseño de componentes tradicionales
- 1.9.5. Desarrollo basado en componentes
- 1.9.6. Patrones de diseño
- 1.9.7. Diseño de software basado en patrones
- 1.9.8. Patrones arquitectónicos
- 1.9.9. Patrones de diseño en el nivel de componentes
- 1.9.10. Patrones de diseño de la interfaz de usuario

1.10. Calidad del software y administración de proyectos
- 1.10.1. Calidad
- 1.10.2. Calidad del software
- 1.10.3. El dilema de la calidad del software
- 1.10.4. Lograr la calidad del software
- 1.10.5. Aseguramiento de la calidad del software
- 1.10.6. El espectro administrativo
- 1.10.7. El personal
- 1.10.8. El producto
- 1.10.9. El proceso
- 1.10.10. El proyecto
- 1.10.11. Principios y prácticas



## **Módulo 2.** Ingeniería del Software Avanzada

2.1. Introducción a las metodologías ágiles
- 2.1.1. Modelos de proceso y metodologías
- 2.1.2. Agilidad y procesos ágiles
- 2.1.3. Manifiesto ágil
- 2.1.4. Algunas metodologías ágiles
- 2.1.5. Ágil vs. tradicional

2.2. Scrum
- 2.2.1. Orígenes y filosofía de Scrum
- 2.2.2. Valores de Scrum
- 2.2.3. Flujo del proceso Scrum
- 2.2.4. Los roles de Scrum
- 2.2.5. Los artefactos de Scrum
- 2.2.6. Los eventos de Scrum
- 2.2.7. Las historias de usuario
- 2.2.8. Extensiones de Scrum
- 2.2.9. Estimaciones ágiles
- 2.2.10. Escalado de Scrum

2.3. Programación extrema
- 2.3.1. Justificación y visión general de XP
- 2.3.2. El ciclo de vida en XP
- 2.3.3. Los cinco valores básicos
- 2.3.4. Las doce prácticas básicas en XP
- 2.3.5. Roles de los participantes
- 2.3.6. XP Industrial
- 2.3.7. Valoración crítica de XP

2.4. Desarrollo de software basado en reutilización
- 2.4.1. La reutilización del software
- 2.4.2. Niveles de reutilización de código
- 2.4.3. Técnicas concretas de reutilización
- 2.4.4. Desarrollo basado en componentes
- 2.4.5. Beneficios y problemas de la reutilización
- 2.4.6. Planificación de la reutilización

2.5. Patrones de arquitectura de sistemas y de diseño de software
- 2.5.1. El diseño arquitectónico
- 2.5.2. Patrones arquitectónicos generales
- 2.5.3. Arquitecturas tolerantes a fallos
- 2.5.4. Arquitecturas de sistemas distribuidos
- 2.5.5. Los patrones de diseño
- 2.5.6. Patrones de Gamma
- 2.5.7. Patrones de diseño de interacción

2.6. Arquitectura de aplicaciones en la nube
- 2.6.1. Fundamentos de Cloud Computing
- 2.6.2. Calidad de las aplicaciones en la nube
- 2.6.3. Estilos de arquitectura
- 2.6.4. Patrones de diseño

2.7. Pruebas del software: TDD, ATDD y BDD
- 2.7.1. Verificación y validación del software
- 2.7.2. Las pruebas de software
- 2.7.3. Test Driven Development (TDD)
- 2.7.4. Acceptance Test Driven Development (ATDD)
- 2.7.5. Behavior Driven Development (BDD)
- 2.7.6. BDD y Cucumber

2.8. La mejora del proceso de software
- 2.8.1. La mejora del proceso de software
- 2.8.2. El proceso de mejora de procesos
- 2.8.3. Modelos de madurez
- 2.8.4. El modelo CMMI
- 2.8.5. CMMI V2.0
- 2.8.6. CMMI y Ágil

2.9. La calidad del producto software: SQuaRE
- 2.9.1. La calidad del software
- 2.9.2. Modelos de calidad del producto software
- 2.9.3. Familia ISO/IEC 25000
- 2.9.4. ISO/IEC 25010: modelo y características de calidad
- 2.9.5. ISO/IEC 25012: la calidad de los datos
- 2.9.6. ISO/IEC 25020: medición de la calidad del software
- 2.9.7. ISO/IEC 25022, 25023 y 25024: métricas de calidad del software y de los datos
- 2.9.8. ISO/IEC 25040: evaluación del software
- 2.9.9. El proceso de certificación

2.10. Introducción a DevOps
- 2.10.1. Concepto de DevOps
- 2.10.2. Prácticas principales

## **Módulo 3.** Ingeniería de Requisitos

3.1. Introducción a la Ingeniería de Requisitos
- 3.1.1. La importancia de los requisitos
- 3.1.2. Concepto de requisito
- 3.1.3. Dimensiones de los requisitos
- 3.1.4. Niveles y tipos de requisitos
- 3.1.5. Características de los requisitos
- 3.1.6. La ingeniería de requisitos
- 3.1.7. El proceso de Ingeniería de Requisitos
- 3.1.8. Frameworks para ingeniería de requisitos
- 3.1.9. Buenas prácticas en ingeniería de requisitos
- 3.1.10. El analista de negocio

3.2. Las fuentes de los requisitos
- 3.2.1. La red de requisitos
- 3.2.2. Los stakeholders
- 3.2.3. Los requisitos de negocio
- 3.2.4. Documento de visión y alcance

3.3. Técnicas de elicitación de requisitos
- 3.3.1. La elicitación de requisitos
- 3.3.2. Problemas de la elicitación de requisitos
- 3.3.3. Contextos de descubrimiento
- 3.3.4. Entrevistas
- 3.3.5. Observación y «aprendizaje»
- 3.3.6. Etnografía
- 3.3.7. Workshops
- 3.3.8. Focus groups
- 3.3.9. Cuestionarios
- 3.3.10. Brainstorming y técnicas creativas
- 3.3.11. Medios grupales
- 3.3.12. Análisis de interfaces del sistema
- 3.3.13. Análisis de documentos y «arqueología»
- 3.3.14. Casos de uso y escenarios
- 3.3.15. Los prototipos
- 3.3.16. La ingeniería inversa
- 3.3.17. Reutilización de requisitos
- 3.3.18. Buenas prácticas de la elicitación

3.4. Requisitos de los usuarios
- 3.4.1. Personas
- 3.4.2. Casos de uso e historias de usuario
- 3.4.3. Escenarios
- 3.4.4. Tipos de escenarios
- 3.4.5. Cómo descubrir escenarios

3.5. Técnicas de Prototipado
- 3.5.1. El prototipado
- 3.5.2. Prototipos según su alcance
- 3.5.3. Prototipos según su temporalidad
- 3.5.4. La fidelidad de un prototipo
- 3.5.5. Prototipos de interfaz de usuario
- 3.5.6. Evaluación de prototipos

3.6. Análisis de requisitos
- 3.6.1. El análisis de requisitos
- 3.6.2. Buenas prácticas del análisis de requisitos
- 3.6.3. El diccionario de datos
- 3.6.4. Priorización de requisitos

3.7. Documentación de los requisitos
- 3.7.1. El documento especificación de requisitos
- 3.7.2. Estructura y contenidos de un SRS
- 3.7.3. Documentación en lenguaje natural
- 3.7.4. EARS: Easy Approach to Requirements Syntax
- 3.7.5. Los requisitos no funcionales
- 3.7.6. Atributos y plantillas en forma de tabla
- 3.7.7. Buenas prácticas de especificación

3.8. Validación y negociación de requisitos
- 3.8.1. Validación de requisitos
- 3.8.2. Técnicas de validación de requisitos
- 3.8.3. Negociación de requisitos

3.9. Modelado y gestión de requisitos
- 3.9.1. El modelado de requisitos
- 3.9.2. La perspectiva del usuario
- 3.9.3. La perspectiva de los datos
- 3.9.4. La perspectiva funcional u orientada al flujo
- 3.9.5. La perspectiva del comportamiento
- 3.9.6. La volatilidad de los requisitos
- 3.9.7. Proceso de gestión de requisitos
- 3.9.8. Herramientas para gestión de requisitos
- 3.9.9. Buenas prácticas en la gestión de requisitos

3.10. Sistemas críticos y especificación formal
- 3.10.1. Los sistemas críticos
- 3.10.2. Especificación dirigida por riesgos
- 3.10.3. Especificación formal





## **Módulo 4.** Procesos de Ingeniería del software

4.1. Marco de Ingeniería software
- 4.1.1. Características del software
- 4.1.2. Los procesos principales en Ingeniería del software
- 4.1.3. Modelos de proceso de desarrollo software
- 4.1.4. Marco de referencia estándar para el proceso de desarrollo de software: la norma ISO/IEC 12207

4.2. Proceso Unificado de desarrollo software
- 4.2.1. Proceso Unificado
- 4.2.2. Dimensiones del Proceso Unificado
- 4.2.3. Proceso de desarrollo dirigido por casos de uso
- 4.2.4. Flujos de trabajo fundamentales de Procesos Unificados

4.3. Planificación en el contexto de desarrollo de software ágil
- 4.3.1. Características del desarrollo software ágil
- 4.3.2. Diferentes horizontes temporales de planificación en el desarrollo ágil
- 4.3.3. Marco de desarrollo ágil Scrum y horizontes temporales de planificación
- 4.3.4. Historias de usuario como unidad de planificación y estimación
- 4.3.5. Técnicas comunes para derivar una estimación
- 4.3.6. Escalas para interpretar las estimaciones
- 4.3.7. Planning poker
- 4.3.8. Tipos de planificaciones comunes: planificación de entregas y planificación de iteración

4.4. Estilos de diseño de software distribuido y arquitecturas software orientadas a servicios
- 4.4.1. Modelos de comunicación en sistemas software distribuidos
- 4.4.2. Capa intermedia o middleware
- 4.4.3. Patrones de arquitectura para sistemas distribuidos
- 4.4.4. Proceso general de diseño de servicios software
- 4.4.5. Aspectos de diseño de servicios software
- 4.4.6. Composición de servicios
- 4.4.7. Arquitectura de servicios web
- 4.4.8. Componentes de Infraestructura y SOA

4.5. Introducción al desarrollo software dirigido por modelos
- 4.5.1. El concepto de modelo
- 4.5.2. Desarrollo software dirigido por modelos
- 4.5.3. Marco de referencia de desarrollo dirigido por modelos MDA
- 4.5.4. Elementos de un modelo de transformación

4.6. Diseño de interfaces gráficas de usuario
- 4.6.1. Principios de diseño de interfaces de usuario
- 4.6.2. Patrones de diseño arquitectónico para sistemas interactivos: Modelo Vista Controlador (MVC)
- 4.6.3. Experiencia de usuario (UX User Experience)
- 4.6.4. Diseño centrado en el usuario
- 4.6.5. Proceso de análisis y diseño de la interfaz gráfica de usuario
- 4.6.6. Usabilidad de interfaces de usuario
- 4.6.7. Accesibilidad en interfaces de usuario

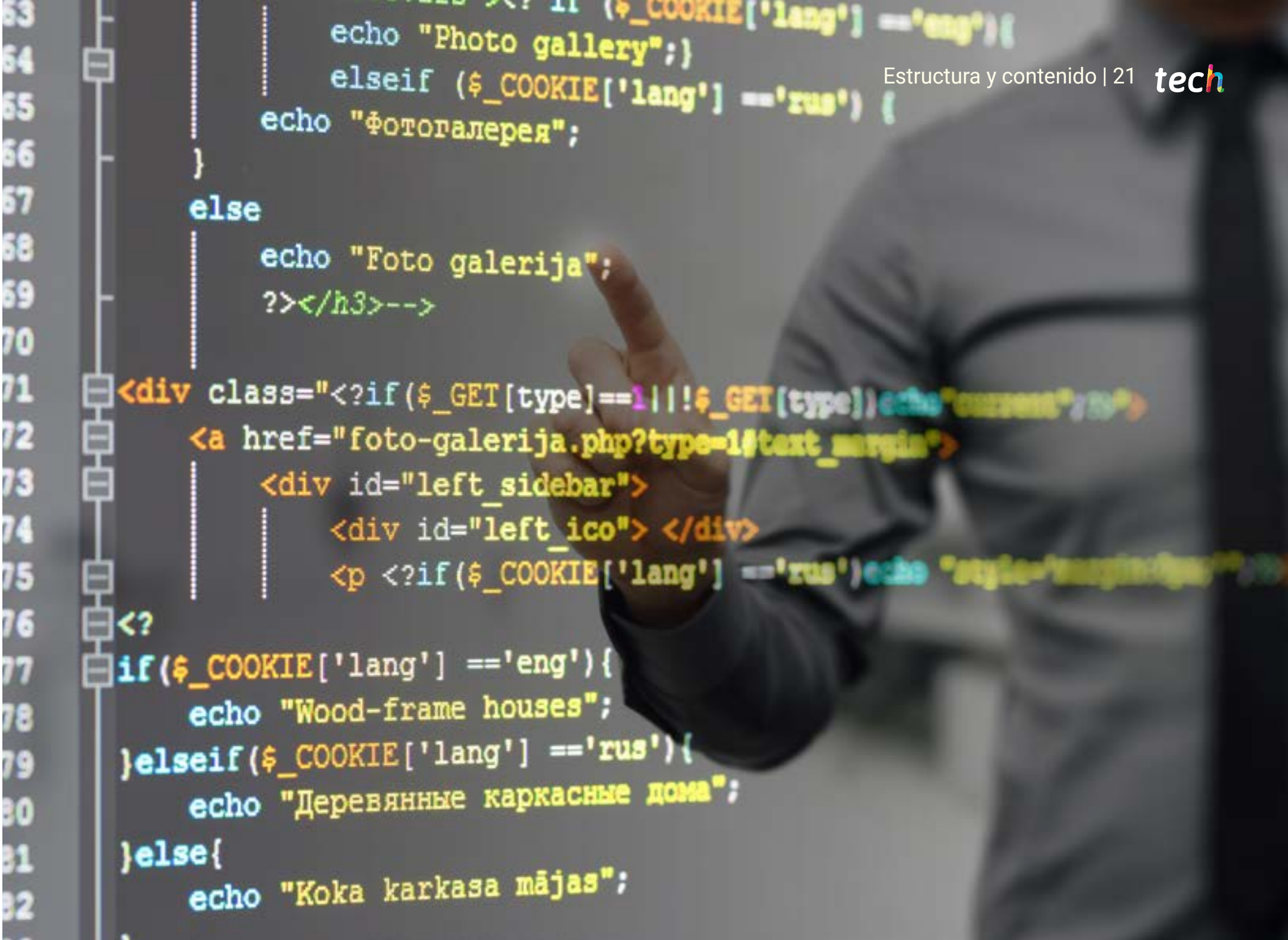
4.7. Diseño de aplicaciones web
- 4.7.1. Características de las aplicaciones web
- 4.7.2. Interfaz de usuario de una aplicación web
- 4.7.3. Diseño de navegación
- 4.7.4. Protocolo de interacción base para aplicaciones web
- 4.7.5. Estilos de arquitectura para aplicaciones web

4.8. Estrategias y técnicas de pruebas software y factores de calidad del software
- 4.8.1. Estrategias de prueba
- 4.8.2. Diseños de casos de prueba
- 4.8.3. Relación coste calidad
- 4.8.4. Modelos de calidad
- 4.8.5. Familia de normas ISO/IEC 25000 (SQuaRE)
- 4.8.6. Modelo de calidad de producto (ISO 2501n)
- 4.8.7. Modelos de calidad de datos (ISO 2501n)
- 4.8.8. Gestión de la calidad del software

4.9. Introducción a las métricas en Ingeniería software
- 4.9.1. Conceptos básicos: medidas, métricas e indicadores
- 4.9.2. Tipos de métricas en Ingeniería software
- 4.9.3. El proceso de medición
- 4.9.4. ISO 25024. Métricas externas y de calidad en uso
- 4.9.5. Métrica orientada a objetos

4.10. Mantenimiento y reingeniería software
- 4.10.1. Proceso de mantenimiento
- 4.10.2. Marco estándar de proceso de mantenimiento. ISO/EIEC 14764
- 4.10.3. Modelo de proceso de reingeniería de software
- 4.10.4. Ingeniería inversa

*Una experiencia de formación única, clave y decisiva para impulsar tu desarrollo profesional”*


echo "Photo gallery";}
elseif ($_COOKIE['lang'] =='rus') {
echo "Фотогалерея";
}
else
echo "Foto galerija";
?></h3>-->
<div class="<?if($_GET[type]==1||!$_GET[type])
<a href="foto-galerija.php?type=1

<p <?if($_COOKIE['lang'] =='rus')
<?
if($_COOKIE['lang'] =='eng'){
echo "Wood-frame houses";
}elseif($_COOKIE['lang'] =='rus'){
echo "Деревянные каркасные дома";
}else{
echo "Koka karkasa mājas";

04

# Metodología

Esta capacitación te ofrece una forma diferente de aprender. Nuestra metodología se desarrolla a través de un modo de aprendizaje de forma cíclica: ***el Relearning***.
Este sistema de enseñanza es utilizado en las facultades de medicina más prestigiosas del mundo y se ha considerado uno de los más eficaces por publicaciones de gran relevancia como el ***New England Journal of Medicine.***





*Descubre el Relearning, un sistema que abandona el aprendizaje lineal convencional, para llevarte a través de sistemas cíclicos de enseñanza: una forma de aprender que ha demostrado su enorme eficacia, especialmente en las materias que requieren memorización"*

## En TECH empleamos el Método del caso

Nuestro programa te ofrece un método revolucionario de desarrollo de tus habilidades y conocimientos. Nuestro objetivo es afianzar tus competencias en un contexto cambiante, competitivo y de alta exigencia.

*Con TECH podrás experimentar una forma de aprender que está moviendo los cimientos de las Universidades tradicionales de todo el mundo"*




*Somos la primera universidad online en español que combina los case studies de Harvard Business School con un sistema de aprendizaje 100% online basado en la reiteración.*



*El alumno aprenderá, mediante actividades colaborativas y casos reales, la resolución de situaciones complejas en entornos empresariales reales.*

## Un método de aprendizaje innovador y diferente

Este programa de Informática de TECH Universidad Tecnológica es un programa intensivo que te prepara para afrontar todos los retos en este área, tanto en el ámbito nacional como internacional. Tenemos el compromiso de favorecer tu crecimiento personal y profesional, la mejor forma de caminar hacia el éxito, por eso en TECH Universidad Tecnológica utilizarás los case studies de la Harvard, con la que tenemos un acuerdo estratégico que nos permite acercarte los materiales de la mejor universidad del mundo.

> *Nuestro programa te prepara para afrontar nuevos retos en entornos inciertos y lograr el éxito en tu carrera"*

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

Ante una determinada situación, ¿qué harías tú? Esta es la pregunta a la que te enfrentamos en el método del caso, un método de aprendizaje orientado a la acción. A lo largo del curso, te enfrentarás a múltiples casos reales. Deberás integrar todos tus conocimientos, investigar, argumentar y defender tus ideas y decisiones.

## Relearning Methodology

Nuestra Universidad es la primera en el mundo que combina los case studies de Harvard University con un sistema de aprendizaje 100 % online basado en la reiteración, que combina elementos didácticos diferentes en cada lección.

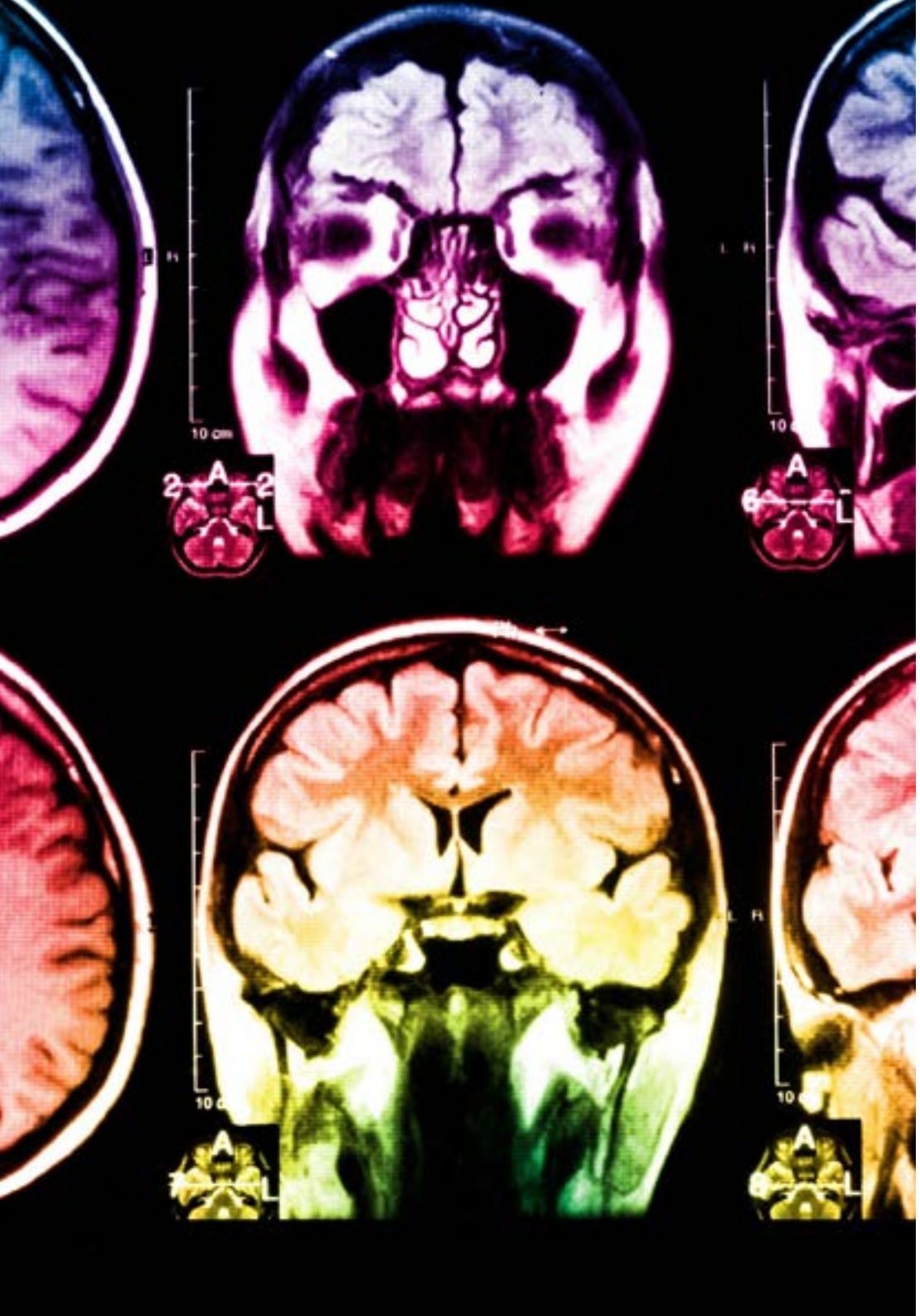
Potenciamos los case studies de Harvard con el mejor método de enseñanza 100 % online: el Relearning.

*En 2019 obtuvimos los mejores resultados de aprendizaje de todas las universidades online en español en el mundo.*

En TECH aprenderás con un na metodología vanguardista concebida para capacitar a los directivos del futuro. Este método, a la vanguardia pedagógica mundial, se denomina Relearning.

Nuestra Universidad es la única en habla hispana licenciada para emplear este exitoso método. En 2019 conseguimos mejorar los niveles de satisfacción global de nuestros alumnos (calidad docente, calidad de los materiales, estructura del curso, objetivos…) con respecto a los indicadores de la mejor universidad online en español.





En nuestro programa el aprendizaje no es un proceso lineal, sino que sucede en espiral (aprendemos, desaprendemos, olvidamos y reaprendemos). Por eso, combinamos cada uno de estos elementos de forma concéntrica. Con esta metodología hemos capacitado a más de 650.000 graduados universitarios con un éxito sin precedentes. En ámbitos tan distintos como la bioquímica, la genética, la cirugía, el derecho internacional, las habilidades directivas, las ciencias del deporte, la filosofia, el derecho, la ingeniería, el periodismo, la historia o los mercados e instrumentos financieros. Todo ello en un entorno de alta exigencia, con un alumnado universitario de un perfil socioeconómico alto y una media de edad de 43,5 años.

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

A partir de la última evidencia científica en el ámbito de la neurociencia, no solo sabemos organizar la información, las ideas, las imágenes, los recuerdos, sino que sabemos que el lugar y el contexto donde hemos aprendido algo es fundamental para que seamos capaces de recordarlo y almacenarlo en el hipocampo, para retenerlo en nuestra memoria a largo plazo.

De esta manera, y en lo que se denomina Neurocognitive context-dependent e-learning, los diferentes elementos de nuestro programa están conectados con el contexto donde el participante desarrolla su práctica profesional.

**En este programa tendrás acceso a los mejores materiales educativos, preparados a conciencia para ti:**

### 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.

### Clases magistrales

Existe evidencia científica sobre la utilidad de la observación de terceros expertos.

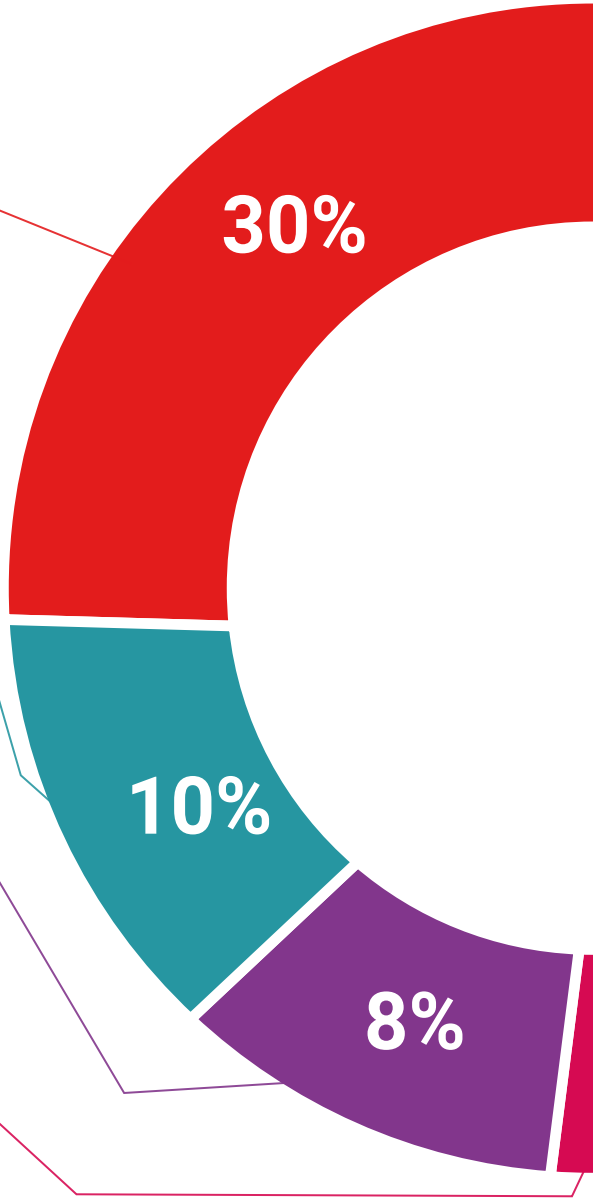
El denominado Learning from an expert afianza el conocimiento y el recuerdo, y genera seguridad en nuestras futuras decisiones difíciles.

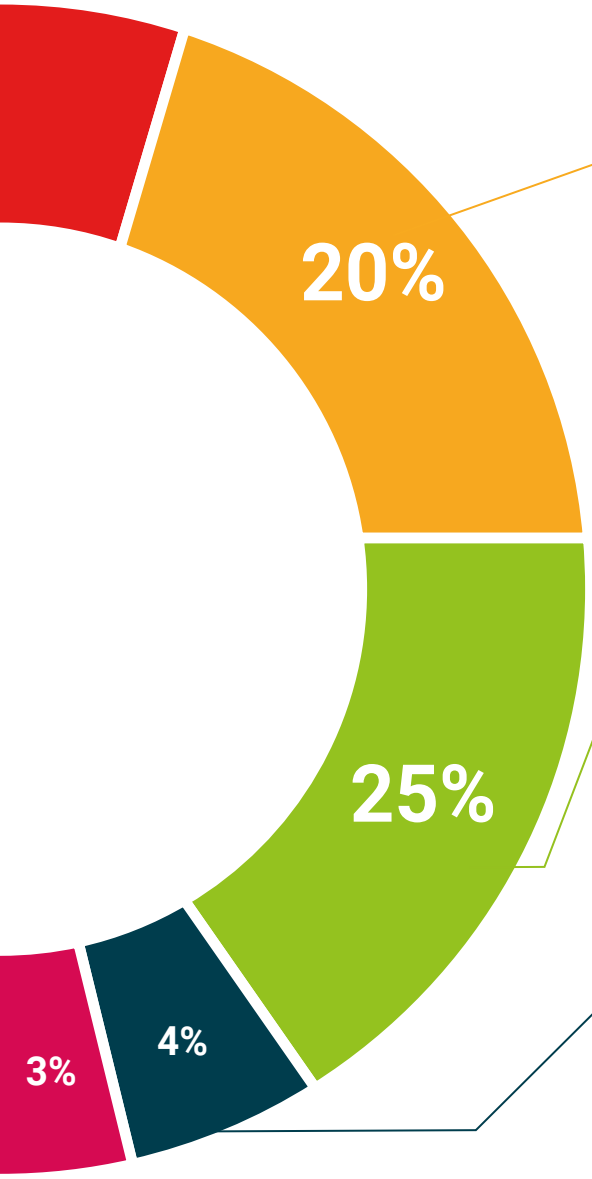
### Prácticas de habilidades y competencias

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.

### 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 cases studies de la materia que se emplean en Harvard. Casos presentados, analizados y tutorizados por los mejores especialistas del panorama internacional.

### 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 para la presentación de contenidos multimedia fue premiado por Microsoft como "Caso de éxito en Europa".

### 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.

05

# Titulación

El **Experto Universitario en Ingeniería del Software** le garantiza, **además de la actualización más rigurosa y actualizada,** el acceso a un título universitario de Máster expedido por la **TECH - Universidad Tecnológica.**





*Supera con éxito esta actualización y recibe tu titulación universitaria sin desplazamientos ni farragosos trámites”*

Este **Experto Universitario en Ingeniería del Software** contiene el programa científico más completo y actualizado del mercado.

Tras la superación de las evaluaciones por parte del alumno, éste recibirá por correo postal con acuse de recibo su correspondiente **Título de Experto Universitario** emitido por la **TECH - Universidad Tecnológica.**

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

Título: **Experto Universitario en Ingeniería del Software**

Nº Horas Oficiales: **600**


tech universidad tecnológica

Otorga la presente

CONSTANCIA

a

C. _____, con documento de identificación nº_____

Por haber superado con éxito y acreditado el programa de

EXPERTO UNIVERSITARIO

en

Ingeniería del Software

Se trata de un título propio de esta Universidad con una duración de 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

*Apostilla de La Haya. En caso de que el alumno solicite que su titulo en papel recabe la Apostilla de La Haya, TECH EDUCATION realizará las gestiones oportunas para su obtención con un coste añadido de 140€ más gastos de envío del título apostillado.

tech universidad tecnológica

# Experto Universitario
## Ingeniería del Software

Modalidad: **Online**

Duración: **6 meses**

Titulación: **TECH - Universidad Tecnológica**

Horas lectivas: **600 h.**

# Experto Universitario
## Ingeniería del Software



tech universidad tecnológica