



Escuela
Superior
de Informática



Universidad de
Castilla-La Mancha

Cursos de Enseñanzas Propias 2018/2019

Ciudad Real, julio de 2018

Escuela Superior de Informática

Universidad de Castilla-La Mancha

Cursos de Enseñanzas Propias 2018/2019

Ciudad Real, julio de 2018

Presentación

Este documento recoge los Cursos de Enseñanzas Propias propuestos por la Escuela Superior de Informática de la Universidad de Castilla-La Mancha para el curso académico 2018/2019.

Se ofrecen siete cursos, cada uno de ellos de veinte horas de duración distribuidas en cuatro sesiones de cinco horas.

Ciudad Real, julio de 2018

Plazos

Para asistir a un Curso de Enseñanzas Propias es preciso preinscribirse, ser admitido y matricularse. Los plazos oficiales para realizar estos trámites son los siguientes:

- **Preinscripción.** Desde quince días lectivos antes de que empiece el curso hasta nueve días lectivos antes.
- **Publicación de lista de admitidos.** Ocho días lectivos antes del inicio del curso.
- **Matrícula.** Desde el séptimo día lectivo anterior al inicio del curso hasta dos días lectivos antes.

A estos efectos, también contabilizan los días de exámenes oficiales de enero, pero nunca los sábados.

Toda la gestión, tanto de preinscripción como de matrícula, se debe gestionar a través de la página de cursos web (<https://cursosweb.uclm.es/>)

Coste

Según acuerdo de la Comisión de Estudios y Programas, la matriculación en los Cursos de Enseñanzas Propias tiene los siguientes costes:

Tipo de estudiante	Precio de matrícula
De la UCLM	80,00 €
Antiguos estudiantes de la ESI	150,00 €
Personal de la UCLM	150,00 €
Ajenos a la UCLM	250,00 €

Procedimiento de admisión

Según acuerdos de Junta de Centro, sobre el total de los estudiantes inscritos, el director del curso seguirá los siguientes criterios de selección para confeccionar la lista de admitidos:

- En primer lugar, estudiantes de todas las titulaciones que se imparten en la ESI, priorizando a los estudiantes de cursos superiores.
- En segundo lugar, el resto de candidatos.

En cada curso se reservarán, no obstante, cinco plazas para candidatos que no sean estudiantes de la Escuela Superior de Informática.

Lugar

Todos los cursos tendrán lugar en las instalaciones de la Escuela Superior de Informática (Edificio Fermín Caballero), Paseo de la Universidad nº 4, 13071-Ciudad Real. La centralita de la Universidad es el 926.295300.

Asistencia

Para obtener el certificado de asistencia, es necesario asistir, al menos, al 80% de las clases.

Reconocimiento de créditos

Para los estudiantes de cualquier grado de la Universidad de Castilla-La Mancha, por cada curso de enseñanzas propias se reconocerá 1 ECTS, siempre y cuando cumplan los criterios de asistencia y entrega de documentación que, para tal fin, se especifiquen en cada curso.

Fechas previstas de celebración de los Cursos de Enseñanzas Propias

CURSOS	Informática forense y evidencia digital	9, 10, 16 y 17 de noviembre de 2018
	Movilidad y chatbots (Avanttic)	9, 16, 23 y 30 de noviembre de 2018
	LaTeX esencial para preparación de Trabajo Fin de Grado, Tesis y otros documentos académicos	1, 8, 15, 22 de febrero de 2019
	Ruby on Rails	1, 8, 15, 22 de febrero de 2019
	Acercamiento de Oracle a los Microservicios (Avanttic)	11, 18 y 25 de febrero y 4 de marzo de 2019
	Introducción a MicroPython para desarrollos IoT	1, 8, 15, 22 de marzo de 2019
	Desarrollo web full-stack con MERN	29 de marzo y 5, 12, 26 de abril de 2019

Índice de Cursos

Informática forense y evidencia digital.....	12
Movilidad y chatbots (Avanttic).....	14
LaTeX esencial para elaboración de TFG, tesis y otros documentos académicos.....	16
Ruby on Rails.....	19
Acercamiento de Oracle a los Microservicios (Avanttic)	21
Introducción a MicroPython para desarrollos IoT	22
Desarrollo web full-stack con MERN.....	25

Informática forense y evidencia digital

Director/a: Juan Carlos López

Secretario/a: M^a José Santofimia

Profesorado:

M^a José Santofimia, David Villa, Félix Jesús Villanueva y Juan Carlos López

Número de estudiantes:

Mínimo: 10

Máximo: 40

Fechas de Realización:

9, 10, 16 y 17 de noviembre de 2018

Descripción:

El curso está enfocado a ofrecer unas nociones generales de la parte técnica y jurídica para que tanto informáticos como juristas puedan enfocar con garantías la fase probatoria con prueba electrónica de un proceso judicial.

Cuestiones como la certificación de contenidos publicados en redes sociales (WhatsApp, Twitter, Facebook, etc.)

Justificación:

Las nuevas tecnologías juegan un papel cada vez más importante en el ámbito jurídico, no sólo porque muchos actos ilícitos recurren al uso de estas herramientas como medio para cometer el acto ilícito sino porque, como elementos de prueba en los procesos judiciales, es cada vez más común encontrar elementos como e-mails, conversaciones de WhatsApp, tweets, etc.

Las diferencias entre la evidencia clásica y la evidencia digital obliga tanto a informáticos como a juristas a tener unas nociones básicas de las implicaciones que tiene el uso de una prueba electrónica (o evidencia digital) en un proceso judicial. Se debe asegurar que las garantías procesales referidas a la prueba se mantienen también cuando la prueba es electrónica. Si un correo electrónico se utiliza como prueba, éste no puede ser aportado al procedimiento como un documento impreso, ya que esto no garantiza el que el texto no ha sido manipulado y que por lo tanto es una prueba fidedigna. Esto mismo ocurre con capturas de pantalla, imágenes, o cualquier otro tipo de archivo electrónico.

Tanto los profesionales de la informática forense como los del ámbito jurídico deben conocer las metodologías de análisis forense, herramientas y fundamentos jurídicos básicos que garanticen que el proceso de recogida de evidencias digitales se realiza con garantías de validez y eficacia, así como su posterior análisis y evaluación una vez aportada como prueba en un proceso.

Programa del curso

1. Introducción

- 1.1. Breve introducción al panorama tecnológico actual: redes sociales, Internet de las cosas o IoT, computación en la nube, GDPR, firma electrónica, blockchain, etc.

- 1.2. Qué es la seguridad informática y cuáles son sus principales objetivos
 - 1.3. Cuando la informática y el derecho se cruzan por el camino
2. Almacenamiento y Recuperación de Información
 - 2.1. Tipos de memoria y papel que estas juegan en el funcionamiento de un computador
 - 2.2. Recuperación de archivos borrados y borrado seguro
 - 2.3. Cifrado de almacenamiento
 - 2.4. Registro de actividades del usuario (registro, cookies, historial del navegador, metadatos, etc.)
3. Comunicación y transferencia de información
 - 3.1. Servicios básicos de la red (DNS, correo electrónico, WWW, etc.)
 - 3.2. Redes inalámbricas (WPS, riesgos redes abiertas y fake-AP,)
 - 3.3. Autenticación, privacidad y anonimización (cifrado de las comunicaciones, certificados, redes Tor, etc.)
4. Evidencia digital
 - 4.1. Validez y eficacia procesal de la prueba electrónica
 - 4.2. La prueba electrónica envenenada
 - 4.3. Recogida de evidencias y la cadena de custodia en la evidencia digital
 - 4.4. Terceros de confianza
 - 4.5. El papel del informe pericial
5. Casos de estudio
 - 5.1. Suplantación de identidad
 - 5.2. Destrucción de información (Discos duros caso Bárcenas)
 - 5.3. Conversaciones de WhatsApp (Caso La Manada)
 - 5.4. Consecuencias legales tras ser objeto de un cibertataque(WannaCry)

Movilidad y chatbots (Avanttic)

Director/a: Macario Polo Usaola

Secretario/a: Macario Polo Usaola

Profesorado:

Rubén Rodríguez (Avanttic)

Número de estudiantes:

Mínimo: 10

Máximo: 20

Fechas de Realización:

9, 16, 23 y 30 de noviembre de 2018

Descripción:

En este curso exploraremos la plataforma de movilidad de Oracle, Oracle Autonomous Mobile Cloud enterprise (AMCe) , que nos permite integrar diferentes sistemas de manera sencilla, además aprenderéis a construir vuestro propio asistente virtual, dotado de inteligencia artificial y que consuma las distintas APIs creadas dentro de AMCe.

Justificación:

Un chatbot es un sistema capaz de interpretar lenguaje natural y, en función de la petición del interlocutor, tomar las acciones que sean necesarias. Múltiples empresas ofrecen a sus clientes funcionalidades diversas mediante chatbots en sus servicios de asistencia telefónica o en sus páginas web. La tendencia es creciente (compañías de telecomunicaciones, de viajes, energéticas, etcétera), por lo que este curso es sumamente interesante para nuestros alumnos, pues se introducirán en tecnologías muy demandadas por el mercado.

Información adicional

Durante las clases, la exposición de la teoría se irá complementando con el desarrollo de un chatbot en equipos de 2 o 3 personas.

Programa del curso

1. Oracle Autonomous Mobile Cloud enterprise (AMCe)

1.1. Overview

1.2. AMCe Core

1.2.1. Mobile Backends

1.2.2. Connectors

1.2.3. Custom APIs

1.2.4. Platform APIs

1.3. Bots

1.3.1. Introduction

1.3.2. Dialog Flow

1.3.3. Custom Components

1.3.4. Channels

2. Reto: Construid vuestro chatbot en equipos de 2 o 3 personas

LaTeX esencial para elaboración de TFG, tesis y otros documentos académicos

Director/a: Jesús Salido Tercero

Secretario/a: Jesús Salido Tercero

Profesorado:

Jesús Salido Tercero

Número de estudiantes:

Mínimo: 15

Máximo: 40

Fechas de Realización:

1, 8, 15, 22 de febrero de 2019

Justificación:

Este curso pretende ser una guía rápida de soluciones a los problemas que se presentan en la elaboración de grandes documentos académicos, especialmente el TFG. Su propósito es desmitificar el uso de LaTeX, un poderoso sistema de preparación de documentos, facilitando su uso para aquellos usuarios que no disponen de suficiente tiempo para estudiar los numerosos y espléndidos manuales publicados al respecto. Al finalizar el curso, el alumno será capaz de elaborar con LaTeX la memoria de un Trabajo Fin de Grado (TFG), Trabajo Fin de Máster (TFM), Tesis y otros documentos académicos: informes, memorias, artículos, ponencias, etc. Durante el curso se seguirán los pasos para la elaboración de documentos, desde la instalación de LaTeX y otras herramientas auxiliares (p.ej. gestor de bibliografía), hasta el uso de algunas de las características más potentes que ofrece dicho procesador. Entre las principales fortalezas de LaTeX destacan la elaboración de índices de contenido, figuras y términos, bibliografía, auto numeración de elementos (figuras, tablas, ecuaciones, listados, etc.), gestión de referencias cruzadas y por supuesto, no debe olvidarse que ¡es libre y gratuito!

Este curso no sólo ofrece información sobre la utilización de LaTeX, también persigue allanar el camino previo del usuario novel durante la instalación y configuración de las herramientas informáticas auxiliares que hacen más cómodo y productivo el trabajo con LaTeX.

A quién va dirigido

Todos los miembros de la comunidad universitaria (estudiantes, profesores, investigadores, etc.), tanto de perfiles técnicos como de otras áreas, pueden aprovecharse de las ventajas que ofrece LaTeX.

Requisitos previos

Como requisitos previos sólo se exige conocimientos básicos, a nivel de usuario, del sistema operativo (SO) a utilizar. No se asume la utilización de un SO concreto y pueden cursarlo los usuarios de Windows, Linux, Mac OSX, etc.

Sobre su desarrollo

El curso se desarrollará en un aula de la Escuela Superior de Informática (ESI-C.Real) en la que existirán equipos con el software preinstalado en Linux. Sin embargo, se recomienda que cada asistente traiga su propio equipo portátil en el que quedará el software configurado para el trabajo personal según sus preferencias personales.

Se proporcionará copias electrónicas de apuntes, transparencias y ejemplos del curso. Los miembros de la UCLM tendrán a su disposición un área de consulta en la plataforma de Campus Virtual (Moodle), pero los alumnos externos a la UCLM dispondrán de un repositorio alternativo con el material del curso.

Programa del curso

1. Introducción

- 1.1. Cuestiones iniciales
- 1.2. ¿Cómo funciona LaTeX?
- 1.3. LaTeX frente a los procesadores WYSIWYG
- 1.4. Instalación de LaTeX
- 1.5. Entornos de trabajo y herramientas auxiliares
- 1.6. El primer documento LaTeX
- 1.7. La codificación de caracteres

2. Tipografía y organización de documentos

- 2.1. Introducción a la tipografía
- 2.2. Tipografía con LaTeX
- 2.3. Caracteres especiales
- 2.4. Notas al pie
- 2.5. Justificación del texto
- 2.6. Listas
- 2.7. Secciones del documento
- 2.8. Tabla de contenidos
- 2.9. Hiperenlaces en PDF
- 2.10. Contadores
- 2.11. Referencias cruzadas
- 2.12. Organizando documentos muy grandes

3. Sesión 3: Documentos enriquecidos.

- 3.1. Inclusión de figuras y gráficos
- 3.2. Formatos gráficos recomendados con LaTeX
- 3.3. Creación e inclusión de tablas
- 3.4. Ubicación de figuras y tablas
- 3.5. Creación e inclusión de ecuaciones
- 3.6. Inclusión de listados de programas y otros elementos
- 3.7. Generación de bibliografía

4. Sesión 4: Personalizando documentos.

- 4.1. Plantillas y paquetes

- 4.2. Organización del documento
- 4.3. Portadas
- 4.4. Márgenes y otros espacios
- 4.5. Cabeceras y pies de página
- 4.6. Títulos e Internacionalización
- 4.7. Otras especialidades

Ruby on Rails

Director/a: Manuel Ángel Serrano Martín

Secretario/a: Manuel Ángel Serrano Martín

Profesorado:

Manuel Ángel Serrano Martín

Número de estudiantes:

Mínimo: 10

Máximo: 20

Fechas de Realización:

1, 8, 15, 22 de febrero de 2019

Justificación:

Actualmente, el desarrollo de aplicaciones informáticas ha cambiado radicalmente y cada vez se utilizan metodologías y tecnologías ágiles que permiten conseguir productos rápidamente, de calidad y ampliables y adaptables a las necesidades del negocio.

Ruby on Rails es un completo marco de trabajo open-source para desarrollar aplicaciones web con acceso a bases de datos de acuerdo con la estructura Modelo-Vista-Controlador (MVC, Model-View-Control). Está optimizado para conseguir hacer feliz al programador y conseguir una buena productividad.

Ruby on rails utiliza una arquitectura de tres capas basada en el paradigma Modelo-Vista-Controlador, utilizando Ajax para el componente de la vista, Ruby para el controlador y proporciona un sistema cómodo de acceso a datos.

Ruby on Rails es utilizado por múltiples sistemas de éxito actual como Twitter, Basecamp, Github, Groupon o Shopify. Es altamente demandado en las ofertas de trabajo actuales y presenta un buen complemento a la formación de los estudiantes de grado en Ingeniería Informática.

El presente curso ofrece una introducción al desarrollo ágil de aplicaciones web basadas, con un enfoque principalmente práctico. Este enfoque permite que pueda ser cursado no solo por estudiantes de Ingeniería Informática, si no por estudiantes o titulados de otras ingenierías o carreras científicas que posean unos mínimos conocimientos de programación e informática en general.

Descripción:

Ruby on Rails es un completo marco de trabajo open-source para desarrollar aplicaciones web con acceso a bases de datos de acuerdo con la estructura Modelo-Vista-Controlador (MVC, Model-View-Control). Está optimizado para conseguir hacer feliz al programador y conseguir una buena productividad.

Ruby on rails utiliza una arquitectura de tres capas basada en el paradigma Modelo-Vista-Controlador, utilizando Ajax para el componente de la vista, Ruby para el controlador y proporciona un sistema cómodo de acceso a datos. Ruby on Rails es utilizado por múltiples sistemas de éxito actual como Twitter, Basecamp, Github, Groupon o Shopify.

El curso es eminentemente práctico y está organizado en una serie de sesiones en las que se expondrán los conceptos básicos de cada apartado y se desarrollarán

ejercicios reales que lleven al dominio del sistema de desarrollo de aplicaciones en este entorno.

Programa del curso

1. Introducción
 - 1.1. Ruby
 - 1.2. Ruby on Rails
 - 1.3. Modelo-Vista-Controlador

2. Introducción a Ruby
 - 2.1. Conceptos básicos de Ruby
 - 2.2. Acceso de ficheros
 - 2.3. Acceso a bases de datos
 - 2.4. Interfaces de usuario

3. Ruby on Rails
 - 3.1. Conceptos básicos de Ruby on Rails
 - 3.2. Ajax
 - 3.3. Modelos
 - 3.4. Vistas
 - 3.5. Controladores
 - 3.6. Deployment
 - 3.7. Conceptos avanzados

Acercamiento de Oracle a los Microservicios (Avanttic)

Director/a: Macario Polo Usaola

Secretario/a: Macario Polo Usaola

Profesorado:

Albeto Cubillo (avanttic Consultoría Tecnológica, S.L)

Número de estudiantes:

Mínimo: 10

Máximo: 20

Fechas de Realización:

11, 18 y 25 de febrero y 4 de marzo de 2019

Descripción:

En este curso explicaremos qué son los microservicios y cómo Oracle nos permite integrar otras aplicaciones con ellos, tanto mediante contenedores como a través de la nube. Además, veremos cómo generarlos, exponerlos y coreografiarlos.

Justificación:

Un microservicio es una funcionalidad que se ofrece, normalmente en la nube, para su utilización desde otro sistema. En multitud de grandes sistemas se utilizan microservicios para acceder a sistemas de pago, de autenticación, de reservas de viajes, etcétera. La utilización de microservicios externos permite al desarrollador centrarse en los aspectos más relevantes de su sistema, pues no debe desarrollar las funcionalidades ofrecidas por el microservicio, sino simplemente acceder a ellas mediante una API. En este curso, el alumno aprenderá a desarrollar y exponer microservicios, área muy demandada por el mercado.

Programa del curso

1. Microservicios
 - 1.1. Introducción
 - 1.2. Desarrollo de microservicios
 - 1.3. Despliegue de microservicios
2. Oracle API Platform
 - 2.1. Introducción
 - 2.2. Generación de APIs
 - 2.3. Implementación de seguridad
 - 2.4. Documentación de APIs
3. Oracle Event Hub Cloud Service
 - 3.1. Introducción
 - 3.2. Kafka Cluster
 - 3.3. Coreografía de servicios

Introducción a MicroPython para desarrollos IoT

Director/a: José Bravo

Secretario/a: José Bravo

Profesorado:

Iván González, Jesús Fontecha, Ramón Hervás

Número de estudiantes:

Mínimo: 5

Máximo: 20

Fechas de Realización:

1, 8, 15, 22 de marzo de 2019

Descripción:

El Internet de las Cosas (IoT) permite la interconexión digital de objetos cotidianos. Estos objetos pueden tener embebidos diferentes sensores y mecanismos actuadores, así como diversos circuitos integrados, incluyendo transceptores de comunicación inalámbrica, buses de comunicación, conversores analógico-digitales, etc. La incorporación de microcontroladores hace posible el gobierno de los objetos cotidianos mediante su programación. Cada vez son más los objetos interconectados que nos rodean, así como sus posibilidades de interacción hombre-máquina y máquina-máquina. Desde bombillas LED y termostatos inteligentes, pasando por macetas autónomas con capacidades de autorriego y monitorización de humedad de la tierra, entre otros muchos ejemplos disponibles.

El crecimiento del mercado IoT se debe al progresivo abaratamiento de los circuitos integrados y, particularmente, al menor coste de los dispositivos basados en microcontroladores. Existen ejemplos de integrados muy económicos y populares, como el ESP32 de Espressif Systems, que son perfectos para IoT. El ESP32 es un “Sistema en Chip” (SoC) que permite construir un sistema embebido autónomo en pocos pasos. Integra en un espacio mínimo transceptores Wi-Fi y Bluetooth, conversores analógico-digitales para obtener lecturas de sensores analógicos, multitud de pines digitales e interfaces de comunicación serial para transmitir datos de sensores digitales y activar/desactivar posibles actuadores. Además, cuenta con un microprocesador doble núcleo que habilita la multitarea con varios perfiles de ahorro de energía.

El ESP32 cobra más relevancia, si cabe, al comprobar su flexibilidad de programación y el amplio abanico de firmwares para implementar su funcionamiento. En este curso introductorio se ha optado por MicroPython, una re-implementación de Python 3 para microcontroladores y SoCs. El hecho de poder utilizar un lenguaje de programación con un nivel de abstracción alto, corriendo sobre un sistema embebido limitado es una hazaña de ingeniería, a la par que permite utilizar uno de los lenguajes de programación más expresivos, potentes y fáciles de comprender sobre un sistema integrado.

En este curso introductorio sobre MycroPython para desarrollos IoT se propone la implementación guiada de un dispositivo conectado empleando el SoC ESP32. El sistema embebido diseñado para tal fin además contendrá un relé como mecanismo actuador y una pareja de sensores analógicos, como ejemplos prácticos de sensorización. A lo largo del curso se presentarán diferentes prototipos que irán

escalando en funcionalidad y complejidad. Partiendo de un montaje en protoboard, sobre una tarjeta de desarrollo del ESP32 (aislado y sin conexión a Internet), hasta un prototipo final sobre una placa de circuito impreso y con capacidades de comunicación gracias al protocolo de transporte MQTT, específico para IoT.

Justificación:

Este curso aborda competencias y contenidos del ecosistema IoT que, siendo profesionalmente demandados y en auge, no se tratan de manera específica en el grado de Ing. Informática. Los contenidos propuestos se imparten siguiendo un esquema teórico-práctico con un enfoque puramente pragmático, adoptando la filosofía de “aprender haciendo”. Cada módulo temático comienza con una explicación de los fundamentos teóricos necesarios para acometer el desafío práctico y termina con la elaboración de un prototipo que va escalando en funcionalidad (tanto a nivel de *hardware* como de *software*) a medida que avanza el curso. La introducción al *hardware* del ESP32, su programación con MicroPython, el despliegue del servicio MQTT o las nociones básicas de automatización de diseño electrónico son las competencias en el marco de desarrollo de aplicaciones IoT que adquirirán los alumnos que se inscriban a este curso.

Programa del curso

0. Motivación del curso

1. Introducción a MicroPython

- 1.1. ¿Por qué Micro?
- 1.2. ¿Por qué Python?
- 1.3. Sistemas embebidos programables con MicroPython
- 1.4. ESP8266 & ESP32 SoCs

2. En detalle: Desarrollando con placas basadas en el ESP32

- 2.1. *Hardware & Pinout*
- 2.2. Alimentación y consumo eléctrico (*Power management*)
- 2.3. Preparando el *setup* de desarrollo y *flasheando* el *firmware* de MicroPython
- 2.4. Implementando *Hello Word!* y otros *scripts* sencillos corriendo MicroPython
- 2.5. Implementando Multitarea básica = ESP32 + MicroPython
- 2.6. Sensores + Actuadores + ESP32 + MicroPython = prototipo de dispositivo IoT (ver. 1.0)
- 2.7. Pruebas y evaluación del prototipo de dispositivo IoT (ver. 1.0)

3. Networking con el ESP32

- 3.1. Conexiones inalámbricas a WLAN (modo estación VS modo punto de acceso)
- 3.2. Introducción al protocolo de mensajería MQTT
- 3.3. Configuración *broker* MQTT (*Mosquitto*)
- 3.4. Cliente MQTT básico en MicroPython para el ESP32
- 3.5. Cliente MQTT básico en Python para PC
- 3.6. Integración MQTT en el prototipo de dispositivo IoT (ver. 2.0)
- 3.7. Pruebas y evaluación del prototipo de dispositivo IoT (ver. 2.0)

4. Migrando del protoboard a la Placa de Circuito Impreso (PBC)
 - 4.1. ¿Qué es una herramienta de automatización de diseño electrónico o EDA?
 - 4.2. Presentando el prototipo de dispositivo IoT en PCB (ver. 3.0). Parte 1-3: componentes
 - 4.3. Presentando el prototipo de dispositivo IoT en PCB (ver. 3.0). Parte 2-3: esquema electrónico
 - 4.4. Presentando el prototipo de dispositivo IoT (ver. 3.0). Parte 3-3: placa de circuito impreso
- Pruebas y evaluación del prototipo de dispositivo IoT (ver. 3.0)

Desarrollo web full-stack con MERN.

Director/a: Manuel Ángel Serrano

Secretario/a: Jesús Fontecha

Profesorado:

Ramón Hervás, Iván González, Jesús Fontecha

Número de estudiantes:

Mínimo: 5

Máximo: 20

Fechas de Realización:

29 de marzo y 5, 12, 26 de abril de 2019

Descripción:

El stack o pila MERN permite la creación sólida de aplicaciones web dinámicas de fácil mantenimiento utilizando las siguientes herramientas: MongoDB, Express.js, React y Node, de ahí sus iniciales. Todas estas tecnologías están basadas en Javascript, con lo cual, tener un lenguaje de programación compartido supone un ahorro de tiempo a la hora del desarrollo de las aplicaciones. En este Curso de Enseñanzas Propias, el alumno conocerá todas las tareas clave que puedan ayudarlo a comenzar, aprender, comprender e implementar aplicaciones web completas en MERN. Se le guiará en el proceso de instalación y configuración de todos los requisitos, para poder crear de forma práctica una aplicación web React en el lado del cliente (Front End) y librerías de servicios REST en el lado del servidor (Back End), todo como parte de la pila o el stack MERN.

Justificación:

El desarrollo de aplicaciones web dinámicas es una profesión en auge y cada vez más demandada debido en gran parte a la digitalización de procesos y servicios en nuestra sociedad. En la actualidad, la mayoría de las empresas necesitan en sus plantillas arquitectos y desarrolladores de software con conocimientos y experiencia en el desarrollo de aplicaciones web, tanto de la parte del cliente como de la del servidor (lo que se conoce como “full stack”). En este sentido, existen multitud de lenguajes y tecnologías web, sin embargo, aquellas basadas en el lenguaje Javascript son de las más demandadas y extendidas hoy en día. En este caso, el “stack” MERN supone la conjunción de diferentes herramientas y tecnologías para el desarrollo de aplicaciones web “full stack” profesionales con base Javascript.

A través de la realización de este curso, el alumno obtendrá los conocimientos básicos necesarios para desarrollar y mantener aplicaciones web MERN usando las tecnologías basadas en JavaScript: ExpressJS, NodeJS y ReactJS, así como la interacción con bases de datos no relacionales MongoDB.

Requisitos deseados (no obligatorios)

Conocimiento de fundamentos del diseño y desarrollo web.

Programa del curso

1. Introducción a JavaScript
 - 1.1. Aspectos y fundamentos del lenguaje
2. Introducción a la pila o “stack” MERN
 - 2.1. Introducción y requisitos técnicos. El patrón MVC
 - 2.2. Instalación y configuración de MongoDB y MongoDB Compass
 - 2.3. Instalación de Node.js
 - 2.3.1. El gestor de paquetes npm
3. Creación de un servidor web con ExpressJS
 - 3.1. Introducción y requisitos técnicos
 - 3.2. Enrutamiento y manejadores de rutas
 - 3.3. Uso de motores de plantillas
 - 3.4. Ejecución y depuración de aplicaciones web ExpressJS
4. Construcción de una API REST
 - 4.1. Introducción y requisitos técnicos
 - 4.2. Operaciones CRUD mediante métodos de enrutado en ExpressJS
 - 4.3. Operaciones CRUD en Mongoose. Modelos e instancias de documentos
 - 4.4. Manejo de solicitudes HTTP. Introducción a los clientes REST
 - 4.5. Ejecución y depuración de una API REST
5. Creación de una aplicación web con React
 - 5.1. Introducción y requisitos técnicos
 - 5.2. Elementos y componentes React
 - 5.2.1. Ciclo de vida, propiedades y estado de los componentes
 - 5.2.2. Enrutado de las páginas del Front End
 - 5.2.3. Conexión de la API REST al Front End React
 - 5.2.4. Ejecución y depuración de componentes React