

# MÁSTER AVANZADO EN CONCEPTUALIZACIÓN (UX/UI) Y DESARROLLO WEB

## Horarios:

### Lunes a Jueves

#### Mañanas

09.00 - 12.00 H.

12.00 - 15.00 H.

#### Tardes

16.00 - 19.00 H.

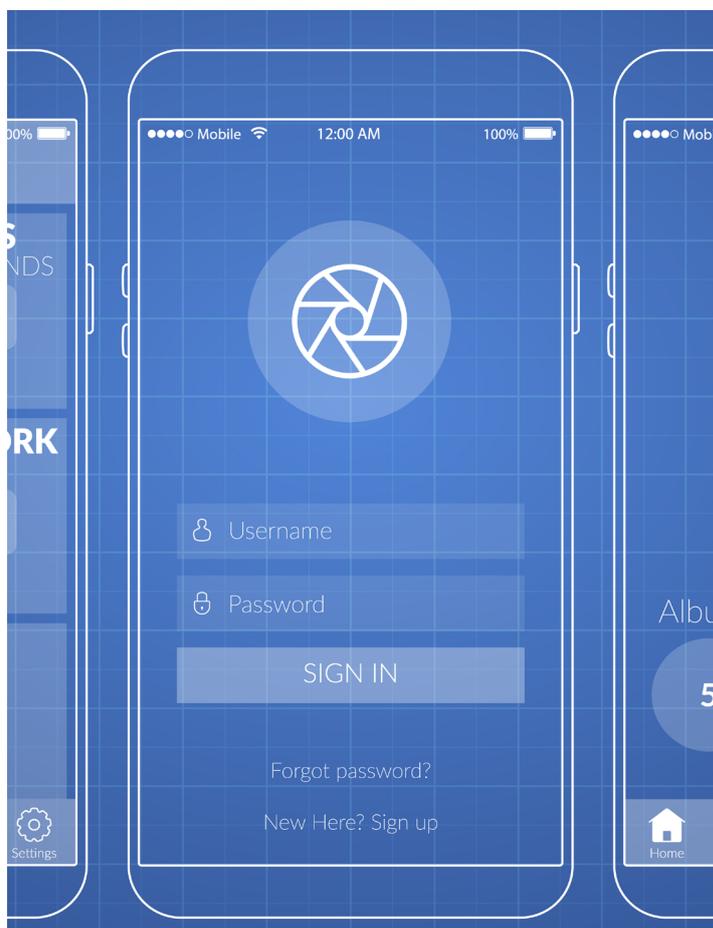
19.00 - 22.00 H.

Pide cita [aquí](#)  
con un asesor de estudios

Consulta precio, ofertas y  
promociones actuales en

[info@cei.es](mailto:info@cei.es)  
[www.cei.es](http://www.cei.es)

# CEI.

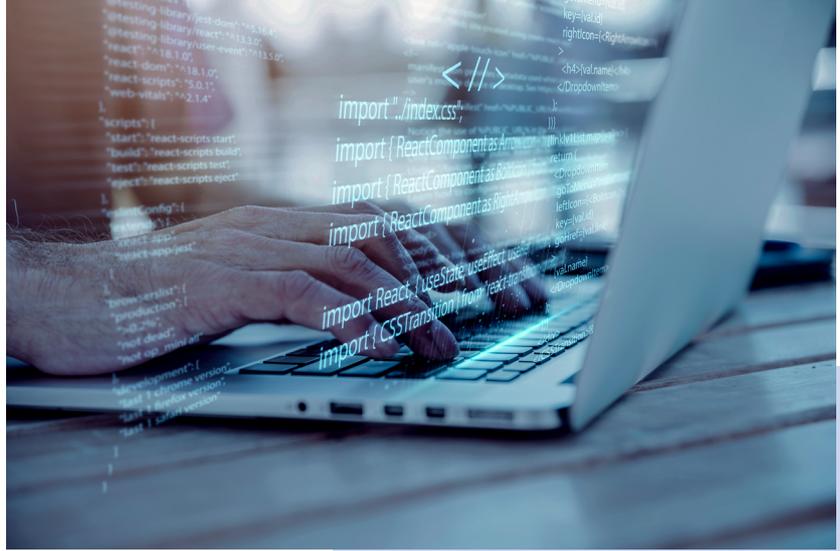


## 540 H. (9 meses)

420 H. Presencial/Online + 120 H. Proyecto final

\* El contenido de este programa es orientativo,  
pudiendo variar el mismo por razones de actuali-  
zación o modificación de los contenidos.

# OBJETIVOS



¿Te imaginas ser capaz de desarrollar una página o aplicación web de principio a fin en tiempo récord? Este Máster Avanzado en Conceptualización (UX/UI) y Desarrollo Web *–full stack–* de CEI cumple con todas las condiciones que un diseñador web necesita hoy en día: dominar la programación web del lado del cliente (*front-end*), del lado del servidor (*back-end*) y, además, el diseño de interfaces y experiencia de usuario (UX/UI).

El programa académico de este Máster Avanzado está compuesto por tres módulos:

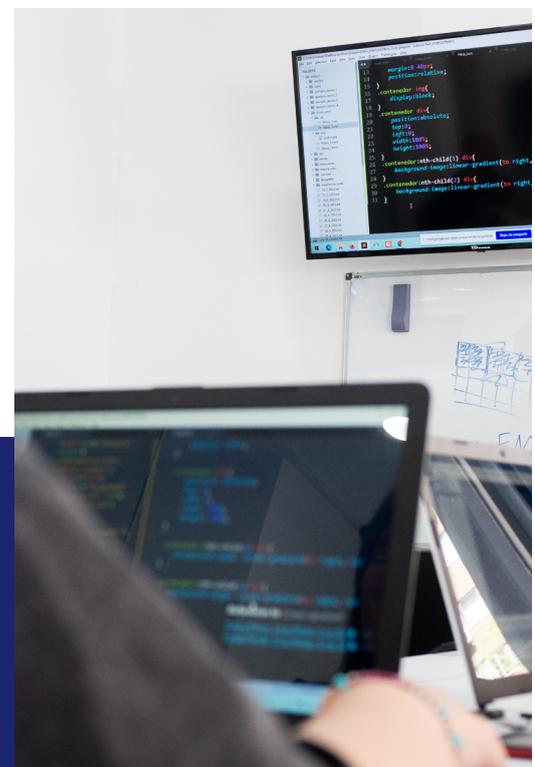
- En el primer módulo abordaremos la conceptualización web, el diseño de interfaces (UI) y la experiencia de usuario (UX). Aprenderás a diseñar interacciones modernas e intuitivas, investigación de usuarios, *stakeholders*, prototipado, arquitectura de la información, combinación de estética y usabilidad, etc.
- Durante el módulo de diseño web con HTML5, CSS3 y Javascript aprenderemos a manejar los lenguajes base de cualquier estructura, maquetación y desarrollo *front-end*: HTML5, CSS3 y JavaScript. Estos lenguajes son esenciales para empezar a darle forma a lo que llamaremos el esqueleto, la “maqueta” o la estructura de cualquier página web, así como algunas interacciones o dinamismos.
- El objetivo del módulo de desarrollo web *Full Stack Development* es brindar al estudiante una base sólida para poder crear aplicaciones web desde la parte del cliente, conocido como *front-end*, hasta la gestión y desarrollo del lado del servidor, lo que llamamos *back-end*.

El *stack* que cubre este módulo es el llamado MERN (MongoDB, Express, React, Node) y lo expande para también utilizar bases de datos relacionales (PostgreSQL y MySQL), sin embargo el enfoque utilizado busca dejar al estudiante preparado con los conocimientos necesarios para poder afrontar los mismos retos en otras combinaciones posibles, inclusive si una de las partes no involucra JavaScript sino otro lenguaje como Python, Go o Rust, entre otros, ya que los conocimientos adquiridos con JavaScript son utilizados para hacer una pequeña introducción a PHP y su ecosistema tecnológico.

# OFICIAL

CEI - Escuela de Diseño y Marketing es un centro de formación reconocido por el Ministerio de Educación para impartir estudios oficiales de grado superior y homologado por las compañías que lideran el sector de las artes digitales y las nuevas tecnologías.

Ser el único centro de formación de España que reúne las siguientes certificaciones es una garantía indiscutible para nuestros alumnos: Adobe Authorized Training Center, Adobe Certified Professional, Apple Authorized Training Center, Autodesk Authorized Training Center, Official Autodesk Certification Center, Maxon Training Provider, Microsoft Certified Educator, Unity Official Training Center, Google Partner... Además, nuestra Escuela ha sido galardonada por los Premios Excelencia Educativa como Mejor Programa Educativo para Inserción Profesional, Mejor Centro de Formación Presencial, Mejor Web de Centro de Formación Presencial y Mejor Centro de Formación Profesional.





## LA ESCUELA

CEI –Centro de Estudios de Innovación– es una escuela de diseño digital fundada en el año 1991 especializada en el sector de las nuevas tecnologías, el diseño y el marketing digital, con sedes en Madrid, Sevilla y Valencia. Nuestros estudios se desarrollan en seis áreas de formación: [diseño gráfico y dirección de arte](#), [diseño y programación web](#), [marketing digital y business](#), [producción, postproducción audiovisual \(VFX\) y motion graphics](#), [diseño de interiores y arquitectura BIM](#), [arte, diseño y programación de videojuegos](#), y [big data y cloud](#).

Ahora sabemos que acertamos cuando, en 1991, apostamos por una enseñanza de calidad como [escuela de diseño, marketing y nuevas tecnologías](#). Nuestra máxima fue, es y seguirá siendo que, el alumno, cuando acabe el periodo lectivo, maneje a la perfección las técnicas impartidas durante su formación. Por esto, la inmensa mayoría de los miles de alumnos que han pasado por nuestras aulas se han incorporado al mercado de trabajo o han mejorado sus condiciones laborales tras terminar sus estudios en CEI.

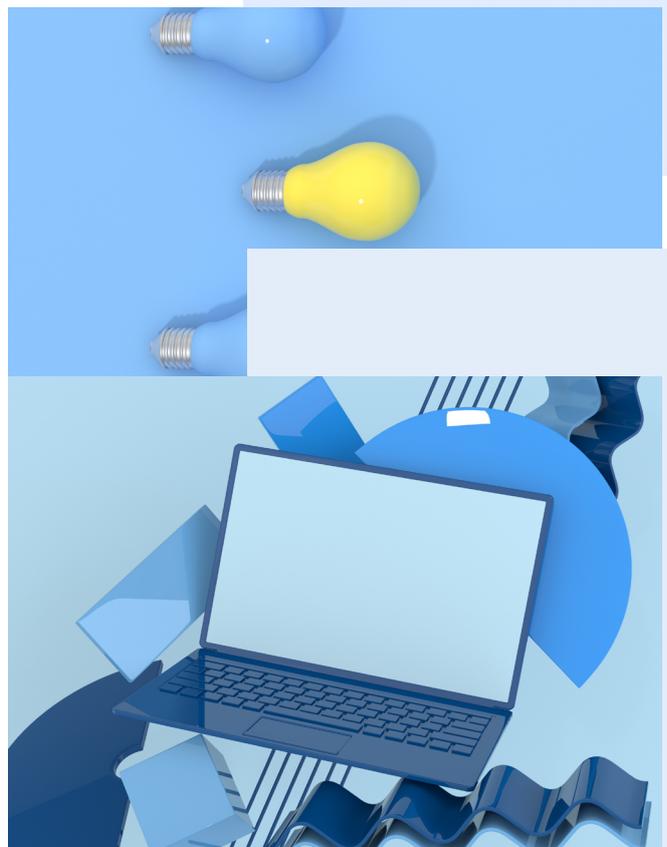
Con más de 30 años en el sector, más de 20.000 alumnos formados y un equipo de profesionales altamente cualificados, CEI pretende potenciar las habilidades de sus alumnos adaptándose constantemente a las fuertes exigencias del mercado laboral de todo el mundo. Es por todo ello que nuestra escuela ha sido galardonada, en los últimos años, por los [Premios Excelencia Educativa como Mejor Centro de Formación Presencial, Mejor Programa Educativo para Inserción Profesional, Mejor Web de Centro de Formación Presencial y Mejor Centro de Formación Profesional](#).

En 2022, CEI unió fuerzas con [Omnes Education Group](#). Omnes Education es una institución privada de educación superior e investigación multidisciplinar que ofrece programas educativos en los campos de la gestión, la ingeniería, la comunicación y los medios digitales y las ciencias políticas. Sus campus se encuentran en Abiyán, Barcelona, Beaune, Burdeos, Chambéry, Ginebra, Londres, Lyon, Madrid, Mónaco, Múnich, París, Rennes, San Francisco, Sevilla y Valencia, repartidos entre 13 escuelas y universidades diferentes. Con más de 40.000 estudiantes cada año y más de 200.000 antiguos alumnos, Omnes ocupa un lugar único en el panorama educativo internacional.

Otro de los puntos fuertes de la escuela es el [equipo de profesores altamente cualificados](#), que ponen a disposición de los estudiantes varios años de experiencia en sus respectivas disciplinas. Un centro de formación homologado ha de comunicar quiénes son sus profesores/as y someterles al control de la compañía propietaria del software sobre el que impartimos formación. Esto garantiza su permanente conocimiento de todas las funcionalidades de las aplicaciones, su contrastada valía y experiencia profesional, así como su indudable capacidad y calidad como docente.

# PROGRAMA

- Fundamentos UX/UI
- *User research*
- Diseño de interfaces (Figma)
- Prototipado
- Introducción al desarrollo web
- HTML
- CSS JavaScript
- AJAX
- NodeJS
- ReactJS
- ExpressJS
- Bases de datos





# MÓDULO

## DISEÑO EN EXPERIENCIA DE USUARIO E INTERFACES (UX/UI) 180 H.

### 01 INTRODUCCIÓN: UX/UI Y FIGMA

- Introducción UX/UI
- Introducción a Figma
- Animación básica en Figma

### 02 DISEÑO E INTERACCIÓN

- Principios de Usabilidad
- Términos y conceptos básicos de Usabilidad
- *UX Writing*
- *Wireframes*

### 03 DISEÑO VISUAL

- Principios de diseño
- Principios de color y tipografía

- Uso de la *Grid* o Retícula
- Accesibilidad

## 04 ARQUITECTURA DE LA INFORMACIÓN

- Árbol de contenidos
- Sistemas de organización
- Sistemas de etiquetado
- Sistemas de navegación
- Sistemas de búsqueda
- Diagrama de flujos

## 05 HEURÍSTICAS

- Análisis Heurístico
- Rediseño e iteración de elementos

## 06 DISEÑO WEB Y APP

- Medidas de dispositivos *web/ipad/mobile/watch*
- Tipografías
- Colores
- *Grid*
- Diseño y prototipado con programas de prototipado avanzado

## 07 PROTOTIPADO AVANZADO

- *Auto layout*
- Componentes
- Variantes
- Variables

## 08 TÉCNICAS DE INVESTIGACIÓN

- *Card Sorting*
- Entrevistas
- Cuestionarios
- Test A/B
- *Benchmarking*

## 9 METODOLOGÍA AGILE

- *Atomic Design*
- *Design Sprint*
- *Design Thinking*

## 10 SISTEMAS DE DISEÑO

- Fundamentos sobre *Design Systems*
- Figma aplicado a Sistemas de diseño. *Foundations*.
- Figma y *Atomic Design*. Componentizado
- Microinteracciones
- Documentación/*hand-off*



# MÓDULO

## DISEÑO WEB CON HTML5, CSS3 Y JAVASCRIPT 180 H.

### 01 INTRODUCCIÓN AL DESARROLLO WEB

- Perfiles en desarrollo web: *Front, Back, Fullstack*
- El diseñador web como programador creativo, el lenguaje de la web
- Herramientas de apoyo
- Editores de texto: factores a tener en cuenta
- Todo diseño web debe ser *responsive*

### 02 HTML: EL CONTENIDO ES EL REY

#### ESTRUCTURANDO NUESTRO CONTENIDO

- Introducción a los lenguajes de etiquetas
- Elementos, etiquetas y atributos
- HTML semántico, el contenido como guía
- Elementos semánticos y elementos estructurales

- Títulos, párrafos, listas, tablas, enlaces e imágenes
- Metodologías HTML y CSS: BEM y SUITCSS

### CREANDO NUESTRO *LAYOUT*

- Relación de las etiquetas estructurales a los típicos “bloques” de un programa de diseño
- Todo es un rectángulo: Modelo-caja
- Unidades de medidas: porcentaje, vh y vw
- *Margin*: separando elementos
- *Padding*, los márgenes internos
- Bordes de una etiqueta
- Posicionamiento estático, o lo que está detrás del flujo normal
- Rompiendo el flujo: *position relative*, *absolute*, *fixed* y *sticky*
- Introducción al diseño *responsive*, *@media queries*

### FORMULARIOS

- Introducción a los formularios
- Elementos de entrada de datos
- Atributos de validación
- Accesibilidad con *Label* en los formularios

## 03 CSS

### CÓMO FUNCIONA EL ESTILO EN UNA WEB O APLICACIÓN

- Flujo de CSS
- Propiedad *display*: *block*, *inline*, *inline-block*, *flex*, *grid*
- Reglas, selectores y propiedades
- Tipos de selectores
- Enlazando HTML y CSS: etiqueta *style*, etiqueta *link*
- Unidades de medida, absolutas y relativas
- Uso del EM y REM como medida responsable
- Inspector de elementos para HTML y CSS

### *LAYOUT* FLEXIBLE CON *FLEX*

- El modelo de caja flexible

- *Flex* vs. flujo normal
- Opciones de *Flex*
- *Flex* y su relación con el diseño *responsive*
- Limitaciones de *Flex*

### *LAYOUT* DE REJILLA: *GRID*

- Cuadrículas a la hora de diseñar una web
- *Grid* vs. *Flex* vs. flujo normal
- Opciones de *Grid*
- *Grid responsive*: 1 línea sin *media-queries*

### LA TIPOGRAFÍA ES MUY IMPORTANTE

- Tipografía web: Google Fonts y tipografías externas
- Propiedades tipográficas
- Contenido dinámico vs. contenido estático y sus consecuencias a la hora de estilizar la tipografía
- Color: hexadecimal, RGB, RGBA
- Color de fondo y color del texto

### LA CAJA DE HERRAMIENTAS GRÁFICAS

- Imágenes de fondo y su diferencia con las imágenes de contenido
- Degradados de color
- Sombras
- Modos de fusión
- Filtros
- Transformaciones
- Transiciones y animaciones

## 04 JAVASCRIPT: DINAMISMO Y FUNCIONALIDAD

### INTRODUCCIÓN A JAVASCRIPT

- ¿Qué es JavaScript?
- Introducción a la programación
- Sintaxis de variables, constantes y funciones
- Variables: datos que pueden cambiar de valor
- Constantes: datos consistentes en nuestro código

## API DEL DOM: LO QUE ASOCIAMOS A JAVASCRIPT

- ¿Qué es una API?
- El DOM
- Distintas formas de capturar elementos del DOM
- Gestión de clases y atributos de HTML
- Añadir y eliminar elementos: HTML dinámico, primera parte

## EVENTOS: AÑADIENDO INTERACCIÓN

- Eventos: *pointers, click, mouseover...*
- Parámetro e: el objeto *event*
- *Callbacks*
- *Intersection Observer*

## AJAX

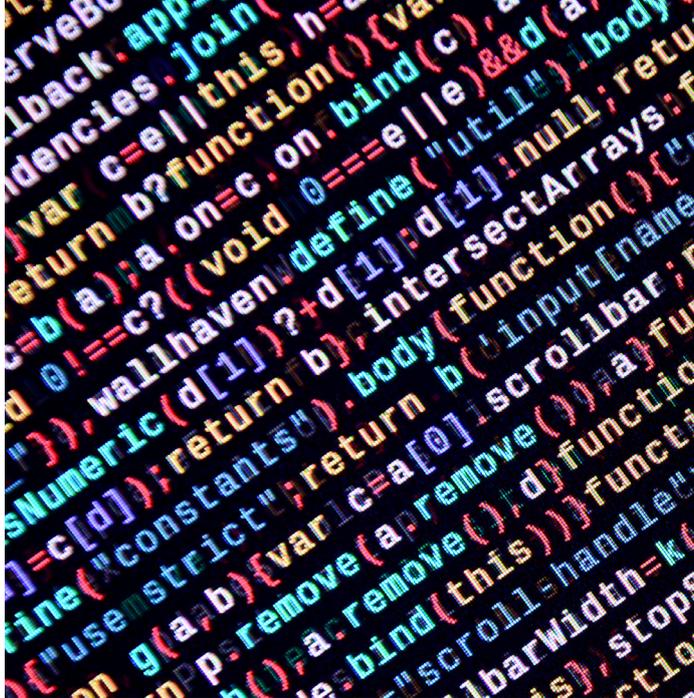
- JSON
- AJAX en el desarrollo web moderno
- *Fetch* en Javascript

## 05 HOSTING

- ¿Qué es un *hosting* y para qué sirve?
- *Hosting* clásico vs. PaaS
- Dominios y *hosting*
- Subir archivo mediante FTP

## 06 GIT Y GITHUB

- La importancia de GIT en empresas
- Usando Github Desktop como control de GIT



# MÓDULO

## DESARROLLO

### WEB

#### *FULL STACK*

#### *DEVELOPMENT*

#### 180 H.

## 01 ESENCIALES PARA FRAMEWORKS

- HTML: metodología SUITCSS, elementos, etiquetas y atributos
- CSS: metodología SUITCSS
- JavaScript: reglas de sintaxis
- Inspector de elementos para HTML, CSS y Javascript (*Debugger*)
- Uso de GIT como sistema de control de versiones

## 02 JAVASCRIPT: BÁSICOS

- *Arrays*: métodos y deconstrucción
- *Arrays*: desestructuración
- Objetos: desestructuración
- Funciones: *arrow function* (parámetros, *return...*)
- Funciones: tipos de funciones (IIFE)
- Funciones asíncronas: Async y Await
- *Timers*: *setTimeout* y *setInterval*
- JS Modules: *Import* y *Export*

## 03 OBJETOS: POO

- Clases e instancias
- Métodos y propiedades
- Herencia

## 04 AJAX: ENVIAR Y RECIBIR DATOS

- HTTP: peticiones y respuestas
- JSON: *parse* y *stringify*
- *Request, Response, Promises*
- *Fetch*: peticiones AJAX utilizando promesas
- *AbortController*

## 05 GUARDANDO INFORMACIÓN

- *Cookies*
- *Local Storage*
- *Session Storage*

## 06 NODE.JS: JAVASCRIPT FUERA DEL NAVEGADOR

- ¿Qué es y para que sirve Node.js?
- Importación y exportación de módulos: ES modules vs. CommonJS
- Módulos nativos
- El sistema de ficheros
- Flujos de datos (*streams*): lectura y escritura
- El módulo HTTP
- Empaquetadores: NPM (*Node Package Manager*)

## 07 REACTJS

- Introducción al uso de librerías/*frameworks*
- ViteJS vs. NextJS

- Conceptos básicos de React: JSX
- Componentes y propiedades
- *Hooks*: *useState*, *useEffect*, *useContext*...
- Eventos
- React Router

## 08 EXPRESSJS

- Estructura básica de una app de Express
- Tipos de métodos: *get*, *post*, *put* y *delete*
- Peticiones asíncronas
- *Middleware*
- *Routing*: Parámetros y *Controllers*
- Configuración del entorno: *Production vs. Developer*

## 09 BASES DE DATOS

- La necesidad de una base de datos
- Distintos paradigmas de bases de datos
- CRUD: *Create, Read, Update, Delete*. Crear, leer, actualizar y borrar
- Bases de datos relacionales vs. documentos
- ORM: *Object-Relational Mapping*
- MongoDB: uso y características
- MongoDB: crear, leer, actualizar y borrar
- MongoDB y ExpressJS: API con MongoDB
- SQL: *Structured Query Language*. Lenguaje de consulta estructurada
- SQL: crear, leer, actualizar y borrar
- SQL y ExpressJS: API con SQL

## 10 PUESTA EN PRODUCCIÓN

- Distintas alternativas para poner en producción nuestros proyectos
- Servidores como repositorios de ficheros estáticos, el modelo tradicional
- Plataforma como servicio (PaaS)
- Infraestructura como servicio (IaaS)