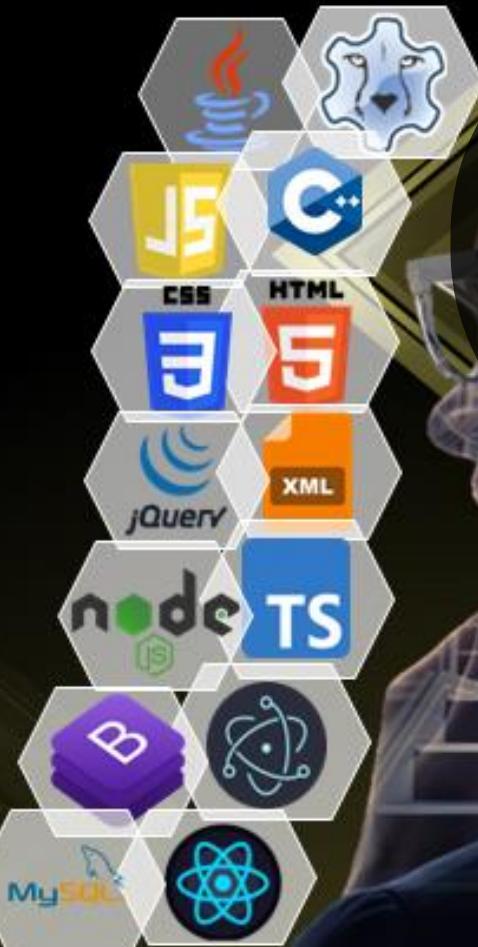


Programación Profesional Desde Cero



Habilidades Clave a desarrollar en el curso

- Programación de software de mediano y alto porte
Aprenderás desde cero a diseñar, planificar e implementar software.
- Programación Web
Sabrás crear aplicaciones Web del lado del cliente y también del servidor.
- Bases de datos relacionales
Diseñarás estructuras de datos basadas en SQL para tus aplicaciones.
- Frameworks de alta demanda
Aprenderás a usar los frameworks y

De la nada al mundo real

El mundo del Desarrollo de Software es extremadamente amplio, por lo que es incluso difícil decidir por dónde empezar a aprender y estudiar, qué lenguaje es mejor, qué herramienta sirve más y por qué, cuál es el camino más adecuado, entre otras cosas. Este curso está diseñado específicamente para solventar estos problemas, estableciendo una hoja de ruta óptima para aprender las habilidades más importantes y generar bases sólidas en el mundo del desarrollo de software.

A través de este camino el estudiante será capaz de dominar la programación en todas sus facetas, conocer los paradigmas más importantes y los lenguajes más demandados y, por sobre todas las cosas, logrará la autonomía suficiente para poder aprender por sí mismo cualquier otra herramienta específica requerida para una tarea concreta. Este es el curso de programación más completo del mercado.



CURSO DE PROGRAMACIÓN PROFESIONAL DESDE CERO

De la nada al mundo real



Sitio Web

www.kaedusoft.edy.uy

Contacto vía WhatsApp o Telegram

(+598) 94 815 035

Mail

consultas@kaedusoft.edu.uy

Contenido

Descripción breve y perfiles.....	5
Objetivos del aprendizaje	5
Perfil del ingreso	6
Perfil del egreso	6
Carga horaria	6
Certificaciones	7
Modalidad de evaluaciones	7
Materiales	8
Programa completo	8
Módulo 0: Introducción y descarga de herramientas (2:13 hs)	8
Módulo 1: Programación estructurada	8
Sección1.1: Entrada y salida estándar de datos (8:12 hs).....	8
Sección 1.2: Selección (4:23 hs).....	9
Sección 1.3: Repetición iterativa (4 hs)	9
Sección 1.4: Arreglos (5 hs).....	9
Sección 1.5: Procedimientos y funciones (subprogramas) (4:15 hs)	9
Sección 1.6: Enumeraciones y registros (3 hs)	9
Sección 1.7: Modularización (3 hs).....	9
Sección 1.8: Manipulación de archivos y persistencia (6:20 hs).....	9
Sección 1.9: Memoria dinámica y punteros (3 hs)	9
Sección 1.10: Modularización con encapsulamiento (10:15 hs).....	10
Sección 1.11: Recursión computacional (1:20 hs).....	10
Módulo 2: Programación orientada a objetos	10
Sección 2.1: Orientación a objetos (6:20 hs).....	10
Sección 2.2: Herencia y polimorfismo (5:15 hs).....	10
Módulo 3: Interfaces gráficas de usuario para Escritorio	10
Módulo 4: Lenguaje HTML	11
Sección 4.1: Elementos básicos y estructuración de página (4 hs).....	11
Sección 4.2: Manipulación avanzada de elementos (1:22 hs)	11
Sección 4.3: Canvas (2:22 hs).....	11
Módulo 5: Hojas de estilo en cascada (CSS)	11
Sección 5.1: Bases de CSS (5hs)	11

Sección 5.2: Diseño con CSS (2 hs)	11
Módulo 6: Programación Web frontend.....	11
Sección 6.1: Document Object Model (DOM) y Browser Object Model (BOM) (2 hs).....	12
Sección 6.2: XML (2 hs)	12
Módulo 7: Bases de datos SQL con MySQL y SQLite	12
Sección 7.1: Preparación (1 hs)	12
Sección 7.2: Bases de SQL y estructuración de datos (5 hs)	12
Sección 7.3: SQL Avanzado (8 hs)	12
Sección 7.4: Diseño de bases de datos.....	13
Módulo 8: NodeJS y Express (10 hs).....	13
Módulosopcionales	13

Descripción breve y perfiles

La habilidad de programar y desarrollar software es de las más demandadas del mercado. Sin embargo suele ser difícil para personas que no conocen mucho (o nada) del tema ingresar a este mercado dada la abrumadora cantidad de tecnologías y ofertas que existen. Este curso está diseñado para que cualquier persona pueda empezar desde cero a programar y adquiera habilidades sólidas y las bases necesarias para postularse a cualquier empleo como programador Junior, escalando así su carrera y ascendiendo a categorías superiores. Del mismo modo el estudiante adquiere las habilidades necesarias para ofrecer sus propios servicios en modalidad Freelancer o emprender su propio negocio en este rubro.

El curso abarca la programación de escritorio y la programación Web, lo cual dota al estudiante de una amplia gama de habilidades que le permiten tener un conocimiento general extremadamente sólido, así como conocimientos específicos que le permitirán aplicar dichas herramientas en la práctica con resultados reales, mensurables y sostenibles en el tiempo.

Asimismo, las bases que el estudiante adquiere le dotan de la capacidad de investigar y aprender de forma autodidacta a utilizar herramientas específicas para resolver problemas concretos que surgen a la hora de trabajar en un proyecto real. Esta habilidad es de extrema importancia y de mucho valor en el mercado, ya que es la norma enfrentarse a cuestiones muy concretas que solo aparecen en cada proyecto y que no son abarcables por ningún curso, por lo cual se requiere la capacidad de buscar información, aprender una herramienta nueva y aplicarla en la resolución de un problema de forma efectiva. Este curso hace mucho hincapié en esta habilidad.

Objetivos del aprendizaje

- **Aprender Programación Estructurada:** Es la primera base que todo programador debe dominar, sin esto no es posible aprender nada más. Este tema se dicta utilizando cuatro (4) tecnologías distintas para que el estudiante adquiera soltura, contraste y conocimiento de los diversos enfoques que existen para el mismo paradigma, siendo éstas: Pascal, C++, Java y JavaScript.
- **Aprender Estructuras de Datos:** El siguiente paso es conocer las estructuras que permiten manipular datos en un programa informático así como los algoritmos para lograr la eficiencia en el uso de recursos del sistema.
- **Aprender Orientación a Objetos:** Sin este conocimiento ningún programador será valioso en el mercado laboral ni podrá hacer uso de las tecnologías disponibles en el mercado. Este tema es la consagración de todos los anteriores.
- **Diseñar software:** Saber programar es solo una parte del desarrollo de software, la cual parte desde el diseño de una aplicación, su estructura, arquitectura y flujos de información. Saber diseñar un programa desde la nada es una habilidad clave que además permite aprender a planificar a largo plazo, establecer metas y objetivos para trabajar luego en su consecución.
- **Diseño de Interfaces Gráficas de Usuario:** El estudiante sabrá diseñar e implementar la interacción del usuario con el software que creará, tanto en aplicaciones de escritorio como en aplicaciones Web utilizando HTML-CSS-JavaScript y frameworks para ello.
- **Uso, diseño y estructuración de Bases de Datos Relacionales:** El estudiante será capaz de trabajar con persistencia de datos utilizando bases de datos SQL.
- **Frameworks esenciales:** El estudiante aprenderá a utilizar frameworks de alta demanda en el mercado como JQuery, TypeScript y React, lo cual le dotará de la soltura para aplicarlos eficientemente así como para aprender cualquier otro que sea necesario en un proyecto concreto.
- **Postularse a empleos reales:** El estudiante será capaz de postularse a ofertas laborales reales y aplicar como Junior para iniciar su carrera en el mercado del desarrollo de software.

Perfil del ingreso

El curso está pensado para aprender desde cero, por lo que no se requiere tener ningún conocimiento previo en programación. Sin embargo sí es necesario que el estudiante tenga un mínimo dominio de informática y ser capaz de descargar programas de internet, guardar archivos y utilizar correo electrónico. El curso enseña paso a paso cómo descargar cada programa y herramienta necesaria para llevarlo a cabo, por lo que el estudiante debe ser capaz de seguir dichos pasos, lo que un conocimiento mínimo del uso de computadoras o de Windows se logra con suma facilidad.

Perfil del egreso

El estudiante obtendrá, a lo largo de todo el curso, distintos perfiles que le dotan de diversas habilidades. En esencia, será un profesional capaz de diseñar e implementar sistemas de software de mediano y alto porte, así como integrarse a equipos de trabajo en proyectos empresariales donde trabaja una gran cantidad de programadores. Asimismo, las habilidades adquiridas le dotarán de la capacidad de ofrecer servicios de desarrollo de software por su cuenta como freelancer ya que tendrá el conocimiento suficiente para crear cualquier aplicación, de escritorio y/o Web por sí mismo. En detalle, el egresado será capaz de:

- Integrarse a equipos de desarrollo de software ya conformados.
- Implementar sistemas de software diseñados por terceros.
- Diseñar estructuras de datos y bases de datos relacionales.
- Trabajar con bases de datos SQL.
- Trabajar en el ámbito frontend.
- Trabajar en el ámbito backend.
- Trabajar en el ámbito fullstack.
- Trabajar como diseñador Web.
- Crear software desde cero.
- Crear y vender sus propias aplicaciones.
- Crear aplicaciones Web.
- Ofrecer servicios de desarrollo de software.
- Investigar y aprender a utilizar cualquier herramienta demandada por un proyecto concreto.
- Dominar varios lenguajes de programación.
- Dominar varios paradigmas de programación.
- Tener un entendimiento amplio de las diversas tecnologías de desarrollo así de cómo y cuándo aplicarlas o por dónde buscar información al respecto.

Carga horaria

El total de horas de clase es 120 hs. Asimismo, el curso ofrece al estudiante varios proyectos prácticos que implicarán una dedicación práctica intensa ya que en esencia esto es un entrenamiento para hacer frente al mundo real y al mercado laboral. Se estima que la carga horaria práctica, incluyendo las consultas docentes, es del 50% del total de horas teóricas, por lo que serían 60 hs más, sumando un total de 180 hs de dedicación al curso.

Horas teóricas	Horas prácticas	TOTAL
120	60	180

El curso es online (remoto) y no tiene tiempos límite, por lo que cada estudiante puede ir a su tiempo y a su ritmo, ajustando sus horarios según su conveniencia. La asistencia docente está siempre disponible vía mail, WhatsApp y Telegram.

Certificaciones

La ruta del curso provee al estudiante de una amplia gama de capacidades, cada una de las cuales se apoya en la anterior pero que, a su vez, es válida por sí misma. Por este motivo, en la medida que el estudiante aprueba diversos módulos obtendrá certificaciones acordes, tal como se detalla:

- ✓ **Programador Estructurado y Orientado a Objetos:** Esta es la base mínima e indispensable para ser considerado valioso en el mercado laboral. Con este conocimiento el egresado ya puede aplicar para ciertos llamados laborales, así como estudiar por su cuenta tecnologías.
- ✓ **Administrador de bases de datos SQL:** El estudiante ha demostrado dominar el uso del lenguaje SQL en la estructuración, gestión y administración de bases de datos en lenguaje SQL.
- ✓ **Administrador MySQL y SQLite:** Se desprende del anterior y especifica las herramientas concretas que el estudiante domina, sin embargo al saber SQL el estudiante es también capaz de dominar otras herramientas basadas en este lenguaje con poco o nada de investigación previa.
- ✓ **Programador Web frontend CSS-HTML-JavaScript:** El Estudiante domina las tecnologías del diseño Web de cara al usuario y al cliente.
- ✓ **Programador Web backend con NodeJS y Express:** El Estudiante domina las tecnologías de desarrollo Web de cara al servidor y sabe utilizar el patrón de diseño MVC, el protocolo HTTP y el uso NodeJS junto a Express.
- ✓ **Programador ElectronJS:** El estudiante domina este framework en concreto. (Opcional)
- ✓ **Programador JQuery:** El estudiante domina este framework en concreto. (Opcional)
- ✓ **Programador TypeScript:** El estudiante domina este framework en concreto. (Opcional)
- ✓ **Programador Bootstrap:** El estudiante domina este framework en concreto. (Opcional)
- ✓ **Programador React:** El estudiante domina este framework en concreto. (Opcional)

Nótese que hay una gran variedad de certificaciones que surgen de un mismo curso, sin embargo solo las 5 últimas son opcionales. Esto se debe a que las primeras abarcan lo mínimo que el estudiante requiere para tener un dominio sólido del desarrollo de software y ser tomado en cuenta en el mercado laboral o tener posibilidades de pasar una entrevista, pero no existe un perfil o título que dé por entendido tal conocimiento por lo que es imperativo desglosarlo en varios certificados. El resto de certificaciones son importantes pero el estudiante no tiene por qué hacerlas todas, sino que puede enfocarse en las que considere necesarias según sus objetivos. Por ejemplo, si pretende postularse a un puesto de trabajo donde se requiere React como herramienta, el estudiante podrá enfocarse en esta herramienta y dejar para después las otras, ya que no son necesarias para el objetivo concreto que necesita. Sin embargo, React, como cualquier otra, requiere del conocimiento de todos los módulos no opcionales del curso. Lo mismo ocurre con el resto de herramientasopcionales.

Modalidad de evaluaciones

El curso provee al estudiante de diversos proyectos obligatorios que deberá programar, los cuales están específicamente diseñados para trabajar los temas dados en el curso y enfrentar al estudiante a casos reales del mercado laboral con el fin de que no solo aprenda a programar, sino que aprenda a lidiar con problemas complejos que enfrentará en cualquier trabajo o al realizar proyectos para clientes propios.

Cada proyecto del curso implica implementar software ya diseñado por los docentes así como diseñar software propio desde cero bajo ciertos requerimientos que deben cumplirse, tal como ocurre en la vida real. Los estudiantes entregarán el código fuente de sus programas, así como la documentación del proceso de desarrollo, diseño y planificación del mismo. Solo cuando todo es aprobado el estudiante recibe la certificación acorde.

Materiales

El curso requiere del uso de aplicaciones y software específico para ser llevado a cabo así como de documentación adicional. El estudiante recibirá todo por parte de KA EduSoft (ECA).

- **Programas necesarios para el curso:** El estudiante verá paso por paso cómo acceder, descargar e instalar las herramientas necesarias para realizar el curso. Ninguna herramienta es de pago, todo el software que se utiliza es gratuito. A saber:
 - Lazarus y FreePascal Compiler
 - NetBeans IDE
 - VisualStudio y VS Code
 - NodeJS
 - Java Runtime Environment (JRE) y Java Development Kit (JDK)
 - Navegador Google Chrome o similar
 - JQuery
 - Electron JS
 - Bootstrap
 - React
 - MySQL Server y MySQL Workbech
 - SQLite y SQLiteSturdio
 - Draw.io
- **Documentación y recursos para proyectos:** Cada vez que se propone un proyecto al estudiante éste podrá descargar la letra del mismo en PDF así como los recursos extra que serán necesarios, tales como partes ya diseñadas del software así como assets y otros materiales.
- **Documentación en línea:** El estudiante deberá aprender a dirigirse a la documentación oficial en línea que todas las tecnologías ponen a disposición de sus usuarios. Durante el curso se enseñará a hacerlo y se brindarán los enlaces pertinentes.
- **Apoyo extra específico:** El estudiante estará en contacto para tener apoyo docente cada vez que lo requiera comunicándose vía mail, WhatsApp o Telegram (según su preferencia). De este modo, si se requiriera, el estudiante puede recibir material de apoyo extra.

Programa completo

Módulo 0: Introducción y descarga de herramientas (2:13 hs)

Objetivo: Descargar e instalar las herramientas necesarias para llevar adelante el curso. Escribir un código básico para asegurarse de que todo funciona correctamente.

1. Descarga e instalación de Lazarus y FreePascal Compiler para uso del lenguaje Pascal.
2. Descarga e instalación de VisualStudio para uso del lenguaje C++.
3. Descarga e instalación de Apache NetBeans IDE para el uso de Java, HTML-CCS-JS.
4. Creación del primer código fuente para comprobar funcionamiento de todas las herramientas.

Módulo 1: Programación estructurada

Objetivo: Aprender el paradigma de *programación estructurada* utilizando el lenguaje Pascal como base para el aprendizaje, trasladando lo aprendido luego a C++, Java y JavaScript.

Sección 1.1: Entrada y salida estándar de datos (8:12 hs)

- | | |
|------------------------------|--------------------------|
| 1. Salida estándar de datos | 6. Comentarios de código |
| 2. Entrada estándar de datos | 7. Cadenas de caracteres |
| 3. Variables y constantes | 8. Secuencias de escape |
| 4. Matemáticas básicas | 9. Inferencia de tipos |
| 5. Tipos primitivos de datos | |

Sección 1.2: Selección (4:23 hs)

- 1. Sentencia IF
- 2. Diagramas de flujo
- 3. Concatenación IF
- 4. Selección múltiple
- 5. Variables booleanas
- 6. Operadores lógicos
- 7. Condiciones compuestas
- 8. Operadores comparativos
- 9. Anidación IF

Sección 1.3: Repetición iterativa (4 hs)

- 1. Sentencia FOR
- 2. Anidación FOR
- 3. Números aleatorios
- 4. Bucle condicional WHILE
- 5. Bucle condicional DO-WHILE
- 6. Banderas booleanas
- 7. Centinelas

Sección 1.4: Arreglos (5 hs)

- 1. Subrangos
- 2. Arreglos lineales
- 3. Búsqueda lineal
- 4. Búsqueda condicional
- 5. Arreglos bidimensionales
- 6. Manipulación de datos
- 7. Acumulación y concatenación
- 8. Uso de depurador
- 9. Recorrido iterativo implícito
- 10. Arreglos dinámicos

Sección 1.5: Procedimientos y funciones (subprogramas) (4:15 hs)

- 1. Procedimientos
- 2. Funciones
- 3. Alcance de identificadores
- 4. Elementos locales y globales
- 5. Sombreado
- 6. Pasaje de parámetros por valor
- 7. Pasaje de parámetros por referencia
- 8. Sobrecarga de identificadores
- 9. Funciones anónimas

Sección 1.6: Enumeraciones y registros (3 hs)

- 1. Enumeraciones
- 2. Registros y estructuras de datos
- 3. Arreglos de registros
- 4. Arreglos con tope
- 5. Estructuras avanzadas

Sección 1.7: Modularización (3 hs)

- 1. Separación de código
- 2. Enlazado HTML
- 3. Caso de estudio: El juego de la vida
- 4. Programación de Buscaminas
- 5. Arquitectura básica de un programa

Sección 1.8: Manipulación de archivos y persistencia (6:20 hs)

- 1. Archivos de texto plano
- 2. Creación, lectura y edición de archivos de texto plano
- 3. Caso de estudio: procesador de textos
- 4. Archivos tipados
- 5. Tipos avanzados de archivos
- 6. Creación de una base de datos manual
- 7. Eliminación de registros
- 8. Creación de un SGBD
- 9. LocalStorage en JavaScript
- 10. Uso de la propiedad InnerHTML

Sección 1.9: Memoria dinámica y punteros (3 hs)

- 1. Conceptos de memoria dinámica
- 2. Punteros a tipos primitivos

- 3. Punteros a registros
- 4. Lista simple encadenada
- 5. Manipulación de nodos de una lista encadenada
- 6. Test de estrés
- 7. Aplicación en JavaScript

Sección 1.10: Modularización con encapsulamiento (10:15 hs)

- 1. Tipos Abstractos de Datos
- 2. Módulos en JavaScript
- 3. Plantillas de Strings en JavaScript
- 4. Expresiones regulares
- 5. Caso de estudio: aplicación de gestión de cursos
- 6. Lista simple encadenada como TAD
- 7. Pila como TAD
- 8. Cola como TAD
- 9. Árbol binario de búsqueda como TAD

Sección 1.11: Recursión computacional (1:20 hs)

- 1. Recursividad
- 2. Pila de llamadas o stack

Módulo 2: Programación orientada a objetos

Sección 2.1: Orientación a objetos (6:20 hs)

- 1. Definición de objetos
- 2. Orientación a objetos
- 3. Definición de clases
- 4. Modificadores de acceso y encapsulamiento
- 5. Listas encadenadas como objetos
- 6. Clases internas
- 7. Operaciones primitivas
- 8. Uso de interfaces predefinidas
- 9. Propiedades GET y SET
- 10. Campos estáticos
- 11. Patrón de diseño SINGLETON
- 12. Programando Batalla Naval
- 13. Objetos literales de JavaScript
- 14. Modularización de JavaScript
- 15. Uso de Prototype

Sección 2.2: Herencia y polimorfismo (5:15 hs)

- 1. Encapsulamiento y herencia
- 2. Herencia y casteo de tipos
- 3. Interfaces
- 4. Generalización de objetos
- 5. Polimorfismo
- 6. Clases abstractas
- 7. Genéricos
- 8. Captura y lanzamiento de excepciones
- 9. Modelo de herencia de JavaScript
- 10. Formato JSON

Módulo 3: Interfaces gráficas de usuario para Escritorio

- 1. El diseño intuitivo al usuario
- 2. Creación de formularios
- 3. Manipulación de componentes visuales
- 4. Uso de foco y tabulación
- 5. Ventanas y cuadros de diálogo
- 6. Elementos avanzados y complejos
- 7. Generación dinámica de componentes
- 8. Modelo de escucha de eventos
- 9. Modelo de escucha de acciones
- 10. Listas filtrables

Llegado este punto, una vez el estudiante finaliza todos los proyectos de forma satisfactoria demostrando su pleno dominio de absolutamente todos los temas, obtendrá el certificado de **Programador Estructurado y Orientado a Objetos**.

Módulo 4: Lenguaje HTML

Sección 4.1: Elementos básicos y estructuración de página (4 hs)

- 1. Reglas horizontales y viñetas
- 2. Enlazado externo
- 3. Enlazado interno
- 4. Tablas
- 5. Formularios
- 6. Barras de progreso
- 7. Elementos meta
- 8. Elementos de estructura
- 9. Formateo de texto
- 10. Uso de imágenes
- 11. Título y favicon

Sección 4.2: Manipulación avanzada de elementos (1:22 hs)

- 1. Uso de documentación oficial
- 2. Workers
- 3. Programación multihilo
- 4. Cuadros de diálogo HTML
- 5. Ventanas internas con IFrame
- 6. Elementos de Video
- 7. Elementos de Audio
- 8. Drag & Drop

Sección 4.3: Canvas (2:22 hs)

- 1. Uso de Canvas
- 2. Arcos, sombras y gráficos
- 3. Mover figuras dinámicamente
- 4. Dibujar dentro de Canvas
- 5. Prototipo de juego
- 6. Scroll de imagen en DIV y en Canvas

Módulo 5: Hojas de estilo en cascada (CSS)

Sección 5.1: Bases de CSS (5hs)

- 1. Uso y aplicación de CSS
- 2. Selectores básicos
- 3. Selectores de ID y clases de estilos
- 4. Fuentes
- 5. Posicionamiento
- 6. Capas y Z-INDEX
- 7. Fondos
- 8. Modelo de cajas
- 9. Alineación
- 10. Menús desplegables
- 11. Barras de navegación
- 12. Sombreado y estilizado
- 13. Bordes de imagen
- 14. Transiciones
- 15. Transformaciones
- 16. Menú contextual
- 17. Tooltips
- 18. Panel tabulado
- 19. Iconos
- 20. Texto multicolumnas
- 21. Unidades de medida
- 22. Modelo de caja flexible

Sección 5.2: Diseño con CSS (2 hs)

- 1. Variables
- 2. Selector de atributos
- 3. Pseudoclases
- 4. Pseudoelementos
- 5. Consultas de medios
- 6. Animaciones
- 7. CSS responsive
- 8. Estilización avanzada

Módulo 6: Programación Web frontend

La programación FrontEnd en realidad es el cúmulo de todas las herramientas anteriores. Este módulo entrega los conceptos finales que permiten el dominio de todo lo visto hasta el momento respecto de JavaScript, HTML y CSS.

Sección 6.1: Document Object Model (DOM) y Browser Object Model (BOM) (2 hs)

- 1. Métodos del DOM
- 2. Remover elementos dinámicamente
- 3. Inserción dinámica de elementos
- 4. Formularios DOM
- 5. Estilos CSS dinámicos
- 6. Selector de consultas
- 7. Nodos del árbol DOM
- 8. Las clases Element y HTMLElement
- 9. DOM sombreado (shadow DOM)
- 10. Creación personalizada de etiquetas HTML
- 11. El uso del BOM

Sección 6.2: XML (2 hs)

- 1. Fundamentos de XML
- 2. XML DOM
- 3. Cargar XML desde archivos
- 4. XML DTD
- 5. XML Schema

Llegado este punto, una vez el estudiante finaliza todos los proyectos de forma satisfactoria demostrando su pleno dominio de absolutamente todos los temas, obtendrá el certificado de **Programador Web FrontEnd CSS-HTML-JavaScript**.

Módulo 7: Bases de datos SQL con MySQL y SQLite

Sección 7.1: Preparación (1 hs)

- 1. Instalación de MySQL Server y Workbench
- 2. Instalación de SQLite y SQLite Studio
- 3. Concepto de cliente-servidor
- 4. Comprobaciones de funcionamiento

Sección 7.2: Bases de SQL y estructuración de datos (5 hs)

- 1. Creación de tablas e inserción de datos
- 2. Carga masiva de datos a una tabla
- 3. Tipos de datos SQL
- 4. Selección filtrada
- 5. Funciones COUNT, SUM, MAX, MIN y AVG
- 6. Gestión de registros y eliminación de bases de datos
- 7. Reestructuración de tablas
- 8. Clave primaria y foránea
- 9. Estructuración compleja
- 10. Inserción y consultas combinadas
- 11. Caso de estudio
- 12. Uso avanzado de estructuras y consultas compuestas

Sección 7.3: SQL Avanzado (8 hs)

- 1. Enumeraciones y conjuntos
- 2. Motores de tablas
- 3. Expresiones regulares en SQL
- 4. INNER JOIN
- 5. Claves foráneas dobles y alias de tablas
- 6. LEFT JOIN
- 7. RIGHT JOIN
- 8. INSERT INTO SELECT
- 9. Variables temporales
- 10. Instrucciones almacenadas
- 11. Volcado de datos
- 12. Transacciones
- 13. Procedimientos almacenados
- 14. Parámetros de salida y de entrada/salida
- 15. Funciones almacenadas
- 16. El tipo BLOB
- 17. Índices
- 18. Vistas
- 19. Disparadores
- 20. Programación en MySQL

Sección 7.4: Diseño de bases de datos

1. Normalización
2. Ciclo de vida de una base de datos
3. Fase de análisis
4. Fase de diseño
5. Fase de implementación
6. Fase de modelo
7. Conexión de una base de datos a un programa creado por uno mismo

Llegado este punto, una vez el estudiante finaliza todos los proyectos de forma satisfactoria demostrando su pleno dominio de absolutamente todos los temas, obtendrá el certificado de **Administrador de bases de datos SQL** y también el de **Administrador MySQL y SQLite**.

Módulo 8: NodeJS y Express (10 hs)

1. Instalación de NodeJS
2. Configuraciones
3. Creación de un servidor HTTP con Express
4. Métodos POST, GET, DELETE y más
5. Uso de Postman para testeo
6. Creación de API REST
7. Despliegue de aplicación en servidor
8. Uso de GIT
9. Integración con SQL
10. Plantillas HTML
11. Creación de cliente de Escritorio

Llegado este punto, una vez el estudiante finaliza todos los proyectos de forma satisfactoria demostrando su pleno dominio de absolutamente todos los temas, obtendrá el certificado de **Programador Web BackEnd con NodeJS y Express**.

Módulos opcionales

En este punto están los módulosopcionales, cada uno de los cuales abarca el dominio de la herramienta en cuestión.

- JQuery
- Electron
- TypeScript
- React
- Bootstrap

Como se describió anteriormente, cada uno de estos módulos otorga una certificación acorde a la herramienta en cuestión. En cada caso la modalidad seguirá siendo la misma teniendo el estudiante que entregar un proyecto que demuestre el uso y dominio de la herramienta según las condiciones dadas por el curso.