



Programa Full Stack JavaScript

De cero a experto



PREPÁRATE PARA

INICIAR EN EL MUNDO DE LA TECNOLOGÍA

Con Enyoi, aprende desde cero habilidades en programación y tecnología. Capacítate con proyectos prácticos, mentores expertos y una metodología que conecta tu formación con oportunidades laborales en el sector digital.

www.enyoi.co

MÓDULO 0: METODOLOGÍAS ÁGILES

Semana del curso	Módulo	Contenidos	Horas de trabajo asincrónico
1	Reglas de juego	Explicación de contrato	0,5
1	Introducción a plataforma enyoi	¿Cómo ingresar al campus virtual? ¿Cómo cambiar mi contraseña? Área personal ¿Qué encuentro en un curso? Calificaciones	0,5
1	Intalación de software requerido	Visual Studio Code e instalación de extensiones	0,5
1	Desarrollo del pensamiento lógico	Descomposición de problemas Pensamiento lógico y condicional Acertijos y juegos de lógica	1
1	Conceptos básicos	¿Qué es Software? ¿Qué es un sistema? ¿Qué es ser desarrollador web?	0,5
1	Metodologías ágiles	Origen de Agile (Manifiesto Ágil). Diferencias entre metodologías tradicionales (cascada) y metodologías ágiles. Principios del Manifiesto Ágil: Valor a las personas, productos funcionales, colaboración con el cliente, y respuesta al cambio.	0,5
1	Scrum	Roles y responsabilidades dentro de un equipo Scrum. Ceremonias de Scrum. Cómo planificar y gestionar Sprints.	0,5
1	Kanban	Creación de un tablero Kanban y definición de columnas. Identificación de cuellos de botella y optimización del flujo.	0,5
1	Prácticas Ágiles para Equipos Remotos	Adaptación de eventos de Scrum a la virtualidad. Herramientas para la gestión ágil de proyectos (Trello, Jira, Miro). Comunicación efectiva en equipos ágiles remotos.	0,5

MÓDULO 1: FUNDAMENTOS DE PROGRAMACIÓN Y BASES DE DATOS

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
1	1	Inducción Enyoi Presentación del coach y cronograma Presentación inicial del proyecto integrador - Sprint 1	2	2	4
1	2	¿Qué es un lenguaje de programación? ¿Qué tipos de lenguajes existen? Variables Tipos de datos Operadores lógicos Estructuras de control de flujo Estructuras de datos	2	2	4
1	3	¿Qué es un algoritmo? Pensamiento algorítmico Tipos de algoritmos	2	2	4
2	4	Pruebas y depuración de algoritmos	2	2	4
2	5	¿Qué es la POO? Objetos Propiedades y métodos Constructores ¿Cómo usar los métodos de un objeto?	2	2	4
2	6	Herencia Polimorfismo Inyección de dependencias	2	2	4
3	7	¿Qué es el levantamiento de requerimientos? Tipos de requerimientos (funcionales y no funcionales) Técnicas básicas de recopilación (entrevistas, encuestas, focus groups) Documentación básica (casos de uso, historias de usuario)	2	2	4

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
3	8	Introducción a bases de datos	2	2	4
		Tipos de bases de datos			
3	9	Modelos de bases de datos	2	2	4
		Introducción a SQL			
4	10	SQL Server	2	2	4
		Manipulación y modificación de datos con SQL			
4	11	Tablas	2	2	4
4	12	Restricciones y desencadenadores	2	2	4
5	13	Optimización de consultas	2	2	4
5	14	Seguridad y control de acceso	2	2	4
		Entrega del proyecto integrador - Sprint 1			
5	15	Sprint Review Meeting 1	2	2	4

MÓDULO 2: FRONTEND

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
1	1	Presentación de los contenidos del módulo y cronograma	2	2	4
		Presentación inicial del proyecto integrador - Sprint 2			
1	2	Historia de la web	2	2	4
		¿Cómo funciona la web?			
1	3	URL - Dominio - Hosting - Protocolo HTTP - DNS	2	2	4
		¿Cómo funcionan los navegadores?			
2	4	Servicios y servidores web	2	2	4
		Páginas estáticas vs. Páginas dinámicas			
2	5	IDE - Editor de código	2	2	4
		Configuración del entorno de trabajo			
2	4	Conociendo el inspector de elementos	2	2	4
		Anatomía de una página web - ¿Qué es HTML?			
2	5	Index HTML y estructura básica	2	2	4
		Anatomía de las etiquetas			

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
2	6	¿Qué es CSS? Reglas de CSS Propiedades CSS	2	2	4
3	7	Modelo de caja Unidades de medida Propiedad position	2	2	4
3	8	Propiedad display Variables CSS	2	2	4
3	9	Flexbox Grid layout	2	2	4
4	10	Responsive Web Design Arquitectura BEM en CSS Introducción a preprocesadores (SASS)	2	2	4
4	11	Preprocesadores SASS - Configuración Preprocesadores SASS - Reglas Preprocesadores SASS - Nesting	2	2	4
4	12	Daily Sprint Meeting	2	2	4
5	13	¿Qué es un sistema de control de versiones? Beneficios de usar control de versiones (historial, colaboración, revertir cambios). Diferencia entre sistemas de control de versiones centralizados y distribuidos. Instalación de Git Configuración inicial de Git (git config): Nombre de usuario (git config --global user.name), Correo electrónico (git config --global user.email), Alias útiles y configuración de editor. git init: Inicializar un repositorio. git status: Ver el estado de los cambios. git add: Añadir archivos al área de preparación (staging). git commit: Confirmar cambios con mensajes descriptivos.	2	2	4
5	14	Diferencia entre el área de trabajo, staging, y el repositorio. Trabajando con Repositorios: git clone: Clonar un repositorio remoto. git pull y git fetch: Sincronizar cambios desde el repositorio remoto. git push: Enviar cambios locales al repositorio remoto.	2	2	4

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
		Concepto de ramas y su importancia en el desarrollo de software. git branch: Crear y listar ramas. git checkout y git switch: Moverse entre ramas. git merge: Combinar cambios de una rama en otra.			
5	15	Estrategias de trabajo con ramas: flujo de trabajo feature-branch. ¿Qué es un conflicto en Git? Cómo identificar y resolver conflictos durante un merge. Buenas prácticas para evitar conflictos comunes. Diferencia entre Git y GitHub. Crear una cuenta y explorar la interfaz de GitHub.	2	2	4
6	16	Creación de un repositorio en GitHub y conexión con un repositorio local. Configuración de git remote para conectar un repositorio local a GitHub. Fork y Pull Request: ¿Qué es un fork y cuándo usarlo?	2	2	4
6	17	Creación de un fork de un repositorio y clonación del mismo. ¿Qué es un pull request y cuál es su propósito? Crear y gestionar pull requests en GitHub. Discusión y revisión de código en pull requests.	2	2	4
6	18	Creación y gestión de issues para reportar bugs o solicitar nuevas funcionalidades. Uso de labels y milestones para organizar tareas. Gestión de proyectos utilizando el tablero de	2	2	4
7	19	¿Por qué usar Javascript? - Estándar ECMA	2	2	4
7	20	Elementos de un lenguaje de programación	2	2	4
7	21	Variables y estructuras	2	2	4
8	22	Operadores: Asignación, comparación y matemáticos	2	2	4
8	23	Estructuras de control	2	2	4
8	24	Métodos con Strings	2	2	4

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
9	25	Métodos Arrays	2	2	4
9	26	DOM - Manejo del DOM - Hosting	2	2	4
9	27	Introducción a API REST	2	2	4
10	28	Animaciones en Javascript	2	2	4
10	29	Herramientas de despliegue	2	2	4
10	30	Sprint Review Meeting 2	2	2	4

MÓDULO 3: BACKEND

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
1	1	Presentación de los contenidos del módulo y cronograma Presentación inicial del proyecto integrador - Sprint 3 Conceptos básicos de IA Principales áreas de la IA Aplicaciones actuales de la IA Ética y desafíos en la IA	2	2	4
1	2	IA para generación y revisión de código	2	2	4
1	3	IA en la automatización y robótica Sistemas de recomendación y personalización Futuro de la Inteligencia Artificial	2	2	4
2	4	¿Qué es Node.js? y ¿Para qué se usa Node.js? Instalación de Node.js y configuración del entorno	2	2	4
2	5	CommonJS vs ESM ¿Qué es NPM y cómo manejar dependencias?	2	2	4
2	6	¿Qué es React.js? ¿Para qué se usa? Sintaxis JSX Instalación y configuración de dependencias	2	2	4
3	7	en React.js ¿Qué son los frameworks?	2	2	4
3	8	¿Qué es Express.js? Iniciando una API REST con Express	2	2	4
3	9	Estructura de un proyecto Node.js con Express Gestión de rutas en Express	2	2	4

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
4	10	¿Qué es Middleware en Express.js? Implementación de middleware personalizado	2	2	4
4	11	CORS (Cross-Origin Resource Sharing) Validación de datos en Express	2	2	4
4	12	Conexión con MySQL ORM con Sequelize Conexión a MongoDB	2	2	4
5	13	Comparación entre bases de datos SQL y NoSQL	2	2	4
5	14	Rutas avanzadas en Express Modularización de controladores Express-jwt: autenticación mediante tokens	2	2	4
5	15	JWT Protección de rutas con JWT	2	2	4
6	16	Manejo de archivos en Node.js Validación y control de tipos de archivos	2	2	4
6	17	Testing en Node.js Debugging y manejo de errores	2	2	4
6	18	Optimización de rendimiento en Node.js Buenas prácticas en el desarrollo con Express	2	2	4
7	19	Seguridad básica en Node.js Manejo de vulnerabilidades comunes	2	2	4
7	20	Implementación de una API REST Parte 1	2	2	4
7	21	Implementación de una API REST Parte 2	2	2	4
8	22	Implementación de una API REST Parte 3	2	2	4
8	23	Daily Sprint Meeting	2	2	4
8	24	Introducción a React Instalación y configuración del entorno de desarrollo Directorio de trabajo	2	2	4
9	25	Componentes en React React Props Ciclo de vida de un componente	2	2	4
9	26	Hooks de efectos y estado (useEffect y useState) React Events	2	2	4
9	27	Eventos personalizados Desestructuración de props y state	2	2	4

Semana del curso	Sesión #	Contenidos	Horas de trabajo sincrónico	Horas de trabajo asincrónico	Intensidad Horaria Total
10	28	React Hooks básicos: useState y useEffect	2	2	4
		React Hooks avanzados: useContext y useReducer			
10	29	React Router DOM	2	2	4
		Navegación dinámica			
10	30	React Forms con Formik	2	2	4
		Validación de formularios con Yup			
11	31	Peticiones con Fetch y Axios	2	2	4
		Manejo de errores y loading states			
11	32	Probando nuestro código	2	2	4
		¿Qué son las pruebas unitarias? Herramientas para ejecutarlas			
11	36	Sprint Review Meeting 3	2	2	4

enyo*i*



¡CONTACTANOS!

 +57 310 4361149

 enyo*i*.co

 enyo*i*.co

 [www.enyo*i*.co](http://www.enyo<i>i</i>.co)



En enyo*i* creemos que la educación en tecnología debe estar al alcance de todos. Por eso, nuestro propósito es formar personas con habilidades prácticas y demandadas, conectándolas con oportunidades reales en el mercado laboral. Trabajamos cada día para cerrar la brecha entre aprendizaje y empleo, impulsando a nuestros estudiantes hacia un futuro lleno de posibilidades. Queremos ser tu aliado en este camino, brindándote las herramientas y el acompañamiento necesario para crecer profesionalmente. ¡Confiamos en ti y en lo que puedes lograr con nosotros!