

Programa

DESARROLLO BACK END

N° créditos: 13

Unidad de competencia

Crear una aplicación moderna con NodeJS y MongoDB, programar con JavaScript del lado del servidor. Trabajar con base de datos NoSQL. Diseñar aplicaciones robustas, rápidas y escalables. Realizar un despliegue en distintas plataformas web.

Sumilla del programa

El programa de especialización en Desarrollo Back-End, es de naturaleza teórico - práctico. Le permitirá al estudiante desarrollar aplicaciones modernas, programando en JavaScript del lado del servidor. Además, aprenderá a trabajar y dominar bases de datos utilizando MongoDB para mejorar la gestión de manera ágil, eficiente, permitiendo que el software sea escalable.

Logro del programa

Al finalizar, el estudiante desarrollará el Back End de una aplicación e-commerce para vender productos de un rubro a elección.

Programación

MÓDULOS	SEMANAS	LECCIONES	CONTENIDOS
MÓDULO 1: INTRODUCCIÓN AL DESARROLLO BACKEND Y CONFIGURAR EL ENTORNO DE TRABAJO. Reconoce los conceptos básicos y el funcionamiento del desarrollo web para el desarrollo del proyecto web.	Semana 1: INTRODUCCIÓN AL DESARROLLO BACKEND Reconoce los lenguajes de programación en tendencia dentro del mundo del Back-end para iniciar con un proyecto web.	<p>1. INTRODUCCIÓN AL BACKEND</p> <p>2. HERRAMIENTAS DE DESARROLLO Y CONFIGURACIÓN DE ENTORNO DE TRABAJO.</p> <p>3. INTRODUCCIÓN A GIT Y EL MANEJO EN NUESTROS PROYECTOS</p>	<ul style="list-style-type: none"> • Stacks tecnológicos • Lenguajes de programación • Aportes de los lenguajes de programación • Principales tendencias visuales en el desarrollo web. <ul style="list-style-type: none"> • Editor de Código • Instalación y entorno de trabajo del editor VS CODE • Extensión para VS CODE. • Creando un proyecto en VS CODE <ul style="list-style-type: none"> • ¿Qué es Git? • Instalación y configuración. • Comandos básicos de Git • Creación de repositorios y ramas. • Flujos de cambios • Archivo gitignore • Repositorio remoto • Configuración de repositorio remoto. • GitHub Pages
		Hackathon 1: El estudiante generará un repositorio sobre el cual podrá practicar conceptos de trabajo en git (creación de un nuevo repositorio, branches, pull-requests y merge).	
MÓDULO 2: FUNDAMENTOS DE PROGRAMACIÓN Crea algoritmos y/o un lenguaje base para dar solución a un problema.	Semana 2: CÓDIGO A PSEUDOCÓDIGO Y USO DE LA TERMINAL Crea algoritmos para dar solución a un problema.	<p>1. USO DE LA TERMINAL Y LÍNEA DE COMANDOS.</p>	<ul style="list-style-type: none"> • ¿Qué es la Línea de comandos? • Comandos básicos de Linux. • Anatomía de un comando. • Trabajando con Directorios y Carpetas. • Creando Archivos y Carpetas

		2. LÓGICA DE PROGRAMACIÓN	<ul style="list-style-type: none"> • ¿Qué es un algoritmo? • Pseudocódigo y Simbología • Estructura de Pseudocódigo Diagrama de flujo • Entrada y Salida Operador de asignación Operadores aritméticos
Hackathon 2: El estudiante podrá usar pseudocódigo, diagramas de flujo o PSeint para crear un algoritmo y determinar cómo funcionará la aplicación, dándole solución a un problema.			
MÓDULO 3: JAVASCRIPT Crea aplicaciones de todo tipo y soluciones web interactivas para trasladar esos conocimientos a cualquier framework JavaScript.	Semana 3: INTRODUCCIÓN A JAVASCRIPT. Aplica el uso de algoritmos haciendo uso de JavaScript para mejorar el funcionamiento de la aplicación.	1. CONCEPTOS GENERALES: SINTAXIS Y VARIABLES 2. CONTROL DE FLUJOS 3. CICLOS E ITERACIONES	<ul style="list-style-type: none"> • Presentación de JavaScript: ¿para qué se usa? • Sintaxis • Uso de la consola. • Variable: ¿qué es y cómo declararla? Asignación y cambio del valor. <ul style="list-style-type: none"> • Condicional: ¿qué es? • Operadores de comparación y lógicos: ¿qué son y cuál es su uso en los condicionales? <ul style="list-style-type: none"> • Ciclos: ¿qué son? Tipos y diferencias entre sí. • Operadores lógicos, ciclos y funciones: ¿cómo combinarlos? • If Else • If Else If • Switch • Ciclo for • Ciclo con for • Ciclo While
Hackathon 3: El estudiante deberá utilizar condicionales y operadores de comparación para ejecutar las mismas instrucciones de código una y otra vez mientras que se cumpla una determinada condición dentro de la aplicación.			
	Semana 4: PROGRAMACIÓN AVANZADA Crea la lógica de programación para reutilizar código dentro de la aplicación.	1. FUNCIONES	<ul style="list-style-type: none"> • ¿Qué es una función? • Parámetros de entrada y de salida • Scope global y local. • Función anónima • Función flecha

		2. OBJETOS	<ul style="list-style-type: none"> • ¿Qué son los objetos y cómo se usan? • ¿Cuáles son sus propiedades y métodos? • Función constructora.
		3. ARRAYS	<ul style="list-style-type: none"> • Métodos de Array • ForEach • Filter • Manipulando strings • Math y date
Hackathon 4: El estudiante deberá generar instrucciones para realizar cálculos o tareas específicas con respecto a la lógica de su aplicación de E-commerce.			
Semana 5: PROGRAMACIÓN ORIENTADA A OBJETOS Emplea POO para la manipulación de datos y reutilización del código.		1. POO (Programación Orientada a Objetos)	<ul style="list-style-type: none"> • Paradigmas de programación • POO y la abstracción • Objetos literales • Asignación de atributos • Notación ES6 • Atributos y métodos
		2. CLASES Y OBJETOS	<ul style="list-style-type: none"> • Redundancia • Función constructora • Clases • Métodos • Herencia
		3. MANEJO Y COMUNICACIÓN DE LOS OBJETOS	<ul style="list-style-type: none"> • Clases para usuarios • Modificando array de objetos
Hackathon 5: El estudiante deberá mejorar la lógica de su aplicación usando clases y objetos para brindar una sintaxis clara y simple en su proyecto.			
Semana 6: MANIPULACIÓN DEL DOM, USO DEL LOCALSTORAGE, Y EVENTOS Aplica eventos en el DOM y localStorage para el almacenamiento de datos de manera local.		1. Storage y JSON	<ul style="list-style-type: none"> • LocalStorage y sessionStorage. • JSON: alcance del formato y situaciones de uso.
		2. DOM	<ul style="list-style-type: none"> • DOM: Definición, alcance y su importancia para operar sobre elementos HTML.
		3. Eventos	<ul style="list-style-type: none"> • ¿Qué son los eventos, para qué sirven y cuáles son los más comunes? • Escuchar un evento sobre el DOM.

			<ul style="list-style-type: none"> • Uso de selectores. • Agregar elementos al DOM.
Hackathon 6: El estudiante podrá manipular del DOM y crear eventos que de la mano con la cual mejorará la lógica del proyecto.			
Semana 7: NodeJS y NPM Usa NodeJS y NPM de lado del servidor en una aplicación para la ejecución de varios comandos como instalar y desinstalar paquetes.	1. NODE JS		<ul style="list-style-type: none"> • ¿Qué es Node.js? • ¿Qué es npm? • Comandos npm • package.json • Paquetes dev • Paquetes prod • Paquetes globales
	2. Package Manager		<ul style="list-style-type: none"> • Gestión de Paquetes.
	3. APIs		<ul style="list-style-type: none"> • Definición de API • Protocolo HTTP • Métodos/Verbos HTTP • Códigos de respuesta HTTP • REST y sus propiedades • Qué es URI y URL • Insomnia y Postman
Hackathon 7: El estudiante debe crear y consumir su API para luego utilizarla del lado del servidor.			
MÓDULO 4: BASE DE DATOS Crea la información de cualquier motor de base a través del lenguaje SQL y NoSQL.	1. BASE DE DATOS RELACIONALES		<ul style="list-style-type: none"> • Diseño Conceptual. • Diseño Lógico. • Diseño Físico.
	2. INTRODUCCIÓN A BASE DE DATOS SQL y NoSQL		<ul style="list-style-type: none"> • Tipos de Base de datos SQL y NoSQL (Postgresql y MongoDB) • Diferencias entre bases de datos • Ventajas y desventajas de NoSQL • Uso de JSON
	3. INTRODUCCIÓN A PostgreSQL		<ul style="list-style-type: none"> • Diseño de base de datos con sentencias SQL • SQL-DDL (Data Definition Language) • SQL-DML (Data Manipulation Language) • Joins.

	Hackathon 8: El estudiante deberá crear el diseño físico y lógico de su base de datos para aplicarlo en su aplicación.	
Semana 9: BASE DE DATOS SQL Utiliza SQL para administrar y acceder a los datos	1. Lenguaje SQL (DDL y DCL)	<ul style="list-style-type: none"> • Conceptos • Crear base y usuario • Tipos de datos • Crear tablas • Modificar y borrar tablas • Otorgar y revocar permisos
	2. Lenguaje SQL (DML y SELECT)	<ul style="list-style-type: none"> • Insertar datos • Actualizar y borrar datos • Restricciones (Constraints) • Clausula Select • Filtros • Ordenamiento.
	3. AGRUPACIONES Y JOINS	<ul style="list-style-type: none"> • GROUP BY • HAVING • LIMIT Y OFFSET • CROSS JOIN • Natural join / Inner join • Left, Right y Full join • Generar Backup.
Hackathon 9: El estudiante deberá crear su base de datos mediante SQL para administrar una Base de Datos y acceder a los datos, asimismo esto le servirá para aplicar esta tecnología a cualquier proyecto.		
Semana 10: BASE DE DATOS NOSQL Utiliza base de datos NoSQL para gestionar Data Documental.	1. OPERACIONES	<ul style="list-style-type: none"> • Listar y eliminar bases de datos • Listar y eliminar colecciones • Listar y eliminar documentos • Leer datos utilizando selectores de consulta • Usando el elemMatch • Agregando índices • Inserción de datos • Subdocumentos y cursores • Modificar documentos sin transacciones • Insertar o modificar documentos con transacciones
	2. AGREGACIONES	<ul style="list-style-type: none"> • Qué son las agregaciones

		<ul style="list-style-type: none"> • Canalización por tuberías • Map Reducers ejemplo simple • Map Reducers ejemplo complejo • Vistas • Métodos de agregación de propósito único
	3. SEGURIDAD Y REPLICACIONES	<ul style="list-style-type: none"> • Seguridad en MongoDB • Autenticación y accesos de control • Verificando permisos • Qué es Replica SET
	4. FRAGMENTACIÓN	<ul style="list-style-type: none"> • Limitantes y la fragmentación • Qué es un clúster fragmentado
	5. MONGOOSE	<ul style="list-style-type: none"> • Qué es Mongoose • Instalación de herramientas • Iniciando la estructura
Hackathon 10: El estudiante deberá generar una Base de Datos usando MongoDB para aplicarla en su proyecto y gestionar Data Documental.		
Semana 11: POSTGRES SQL Crea bases de datos robustas a partir del motor de bases de datos PostgreSQL.	1. PROGRAMACION PROCEDIMENTAL	<ul style="list-style-type: none"> • PgAdmin y Schema • Clausula DISTINCT • Lenguaje procedural y funciones • Creando función descuento con condicionales • Cuando usar procedimientos almacenados y concatenación • Creando función para paginar resultados • Tipos de datos que acepta PostgreSQL • Ciclos For y While • Creando función consulta de ventas
	2. TRIGGERS	<ul style="list-style-type: none"> • Triggers • Preparar función para el trigger • Creando el trigger • Crear tabla y función auditoria

			<ul style="list-style-type: none"> • Trigger de auditoria • Trigger: Inventario y 8unción compras • Trigger: Inventario y 8unción venta • Pruebas de funcionamiento
		3. FORMAS NORMALES	<ul style="list-style-type: none"> • Normalizacion, cálculo de dependencias y claves • Formas normales • Proceso de normalización y ejemplo • Encontrar dependencias funcionales
Hackathon 11: El estudiante deberá generar una base de datos usando PostgreSQL para el desarrollo de una aplicación.			
MÓDULO 5: PROGRAMACIÓN BACKEND Diseña aplicaciones complejas de Back End robustas, rápidas y escalables, dominando diferentes técnicas de comunicación, manejo de procesos distribuidos y control de grandes volúmenes de datos para dominar la gestión de los datos en forma eficiente, ágil y con una gran facilidad de escalabilidad.	Semana 12: SERVIDORES CON JAVASCRIPT Crea desde el inicio el proyecto de Bckend a través de la instalación de un servidor del lado del servidor.	1. ECMASCIPT Y TYPESCRIPT	<ul style="list-style-type: none"> • ECMAScript • Typescript. • Programación Sincrónica vs Asincrónica.
		2. MANEJO DE ARCHIVOS	<ul style="list-style-type: none"> • Callbacks • Promesas
		3. NODEJS	<ul style="list-style-type: none"> • Introducción a NodeJS
		4. ADMINISTRADOR DE PAQUETES - NPM.	<ul style="list-style-type: none"> • Módulos nativos • NPM • Instalaciones globales y locales
	Hackathon 12: El estudiante deberá realizar la instalación de NodeJS así como el manejo de los paquetes para utilizarlos en el desarrollo de su aplicación.		
	Semana 13: EXPRESS Utiliza Express su aplicación del lado del servidor.	1. SERVIDORES WEB	<ul style="list-style-type: none"> • Protocolo HTTP • Swervidor con Express JS • Objeto Request
		2. EXPRESS AVANZADO	<ul style="list-style-type: none"> • Status de protocolo HTTP • API REST • Uso de los métodos POST, PUT, DELETE
		3. ROUTER Y MULTER	<ul style="list-style-type: none"> • Express Router • Express y Middlewares • Multer

		Hackathon 13: El estudiante deberá crear un archivo de form de registro de un usuario para enviar esos datos a una ruta POST.
Semana 14: WEBSOCKETS Aplica WebSockets para obtener una comunicación entre el navegador del usuario y un servidor.	1. EL FUNCIONAMIENTO DE LOS WEBSOCKETS	<ul style="list-style-type: none"> • Bienvenida y definición de WS • Creación de un servidor de WS • Nuestro primer cliente WS • Enviar y recibir JSON
	2. SOCKETS CON EXPRESS CON Sockets.io	<ul style="list-style-type: none"> • Sockets.io • Características • Instalación
Hackathon 14: El estudiante deberá realizar la comunicación entre el cliente y el servidor para así transferir datos de manera eficiente y en tiempo real.		
Semana 15: CRUD EN MONGO Crea un CRUD con ayuda de MongoDB para aplicarlo en un proyecto web.	1. CRUD EN MONGODB	<ul style="list-style-type: none"> • Crea un CRUD en Mongo • Filtros • CRUD-UD
	2. Moongoose en NodeJS	<ul style="list-style-type: none"> • Implementando Mongoose • Creando nuestro primer modelo • Insertando en nuestro modelo • Creando servicios API • Accediendo a los datos y realizando operaciones
Hackathon 15: El estudiante deberá almacenar documentos flexibles mediante MongoDB y analizar la sintaxis y aplicar una búsqueda usando ordenamiento, proyecciones, saltos y límites.		
Semana 16: Cookies, Session & Storage. Crea el login de su aplicación a partir de Cookies, Session y Storage.	1. Cookies	<ul style="list-style-type: none"> • ¿Qué es una Cookie? • Como utilizar una Cookie • Set, Get y Clear.
	2. Session	<ul style="list-style-type: none"> • Identidad del Cliente: Sessions • Session como módulo de Node. • Utilizando Session • Eliminar datos de la Session • Login con Session.
	3. Storage	<ul style="list-style-type: none"> • Storage • Memory Storage • File Storage • Session Storage con Mongo Atlas
Hackathon 16: El estudiante deberá crear un login para el acceso a su aplicación de e-commerce		

	<p>Semana 17: AUTORIZACIÓN Y AUTENTICACIÓN + JWT Crea la protección de las passwords del cliente a partir de la diferenciación entre autenticación y autorización.</p>	<p>1. AUTORIZACIÓN Y AUTENTICACIÓN</p> <p>2. ESTRATEGIAS DE AUTENTICACIÓN POR TERCEROS</p>	<ul style="list-style-type: none"> • Autorización y autenticación • Estrategias de autenticación passport <ul style="list-style-type: none"> • Concepto de Autenticación por Terceros • Autenticación para GitHub.
Hackathon 17: El estudiante deberá crear su aplicación haciendo uso de Autenticación, Autorización y JWT, para mejorar la seguridad de su proyecto.			
MÓDULO 6: FEEDBACK, ASESORÍA Y PRESENTACIÓN DEL PROYECTO FINAL	<p>Semana 18: Proyecto Final</p>	<p>1. Asesorías virtuales</p>	<ul style="list-style-type: none"> • Herramientas para la gestión de proyectos • Recomendaciones • Feedback sobre proyecto final
		<p>2. Asesorías virtuales</p>	<ul style="list-style-type: none"> • Herramientas para la gestión de proyectos • Recomendaciones • Feedback sobre proyecto final
		<p>3. Presentación del Proyecto Final</p>	<ul style="list-style-type: none"> • Temas Presentaciones • Evaluación y Feedback. • Entrega de certificado Idat.

Sistema de Evaluación

EC	EF
Evaluación continua	Evaluación final
Peso	Peso
40%	60%

Referencias bibliográficas

Bibliografía
virtual
básica

Base de datos E-Libro:

- Mozilla (2023) Resources for Developers, by Developers, Mozilla Firefox. Recuperado el 29 de enero de 2023. <https://developer.mozilla.org/es/>
- Mozilla (2023) JavaScript, Mozilla Firefox. Recuperado el 29 de enero de 2023. <https://developer.mozilla.org/es/docs/Web/JavaScript>
- NodeJS (2023) Node.Js. Recuperado el 29 de enero de 2023. <https://nodejs.org/es/>
- MongoDB (2023) MongoDB. Recuperado el 29 de enero de 2023. <https://www.mongodb.com/>