



PLAN FORMATIVO	DESARROLLO DE APLICACIONES FULL STACK JAVASCRIPT TRAINEE V2.0
CÓDIGO PLAN FORMATIVO	PF1163
SECTOR	INFORMACIÓN Y COMUNICACIONES
SUB SECTOR	TECNOLOGÍAS DE INFORMACIÓN
AREA	Computación e informática
SUB AREA	Software
ESPECIALIDAD	Diseño, desarrollo y programación (página web, aplicaciones móviles, video juegos, entre otros)
PERFILES ASOCIADOS	SIN PERFIL DE CHILEVALORA ASOCIADO
NIVEL CUALIFICACION	Nivel 4
MODALIDAD PRESENCIAL / NO PRESENCIAL (ONLINE)	E-LEARNING, TODOS LOS MÓDULOS SON ADAPTABLES A MODALIDAD ONLINE.
DESCRIPCIÓN DE LA OCUPACIÓN Y CAMPO LABORAL ASOCIADO	Este plan está diseñado para desarrollar las habilidades necesarias para construir y mantener piezas de software en lenguaje Javascript y frameworks basados en Javascript, crear aplicaciones web en full stack Javascript y operar una base de datos para almacenar los datos, a partir de especificaciones técnicas dadas que den solución a las problemáticas de la organización, aplicando buenas prácticas de programación con el objeto de lograr un producto con niveles de calidad acordes a las necesidades de la industria. El campo laboral corresponde a Organizaciones, ya sean públicas o privadas, transversal a todas las industrias (retail, banca, salud, minería, manufactura, servicios), que realicen desarrollo, implementación y mantenimiento de software, ya sean productivas, servicios, gubernamentales, o que prestan servicios TI a otras organizaciones, que requieran reclutamiento de desarrolladores trainee para procesos de selección, entrenamiento y ponerlos posteriormente a disposición de los equipos de proyectos de software en un rol Junior o Trainee.
VERSIÓN N°	8
N° RESOLUCIÓN	934
FECHA DE RESOLUCIÓN	06-09-2023

REQUISITOS OTEC	Este plan formativo fue creado en el contexto del programa Talento Digital para Chile, por tanto, deberá ser dictado por Bootcamps que cuenten con rol de OTEC o bien, por OTEC en convenio con Bootcamps ya sea nacionales o internacionales.
INSTRUMENTO HABILITANTE PARTICIPANTE	Sin Instrumento habilitante.
REQUISITOS DE INGRESO AL PLAN FORMATIVO	Licencia de Enseñanza Media completa; Manejo computacional a nivel usuario; Aprobación de test de selección en etapa I y II, consistente en prueba en línea que mide habilidades de raciocinio lógico y comprensión verbal, más un test de aptitud. **Se exceptúan de este requisito, a los postulantes al programa Talento Digital para Chile, que estén cursando cuarto año medio en liceos científico-humanista o en el caso de alumnos de liceos técnico profesional que estén cursando cuarto medio o desarrollando su práctica laboral.”
COMPETENCIA DEL PLAN FORMATIVO	Desarrollar aplicaciones web Full Stack utilizando tecnología Javascript que den solución a las necesidades de la organización, aplicando las buenas prácticas de la industria para obtener un producto con niveles de calidad requeridos.



NÚMERO DE MÓDULOS	NOMBRE DEL MÓDULO	HORAS DE DURACIÓN
Módulo N°1	ORIENTACIÓN AL PERFIL Y METODOLOGÍA DEL CURSO	18,00
Módulo N°2	FUNDAMENTOS DE DESARROLLO FRONT-END	72,00
Módulo N°3	FUNDAMENTOS DE PROGRAMACIÓN EN JAVASCRIPT	60,00
Módulo N°4	PROGRAMACIÓN AVANZADA EN JAVASCRIPT (V2)	48,00
Módulo N°5	FUNDAMENTOS DE BASES DE DATOS RELACIONALES	56,00
Módulo N°6	DESARROLLO DE APLICACIONES WEB NODE EXPRESS	72,00
Módulo N°7	ACCESO A DATOS EN APLICACIONES NODE	64,00
Módulo N°8	IMPLEMENTACIÓN DE API BACKEND NODE EXPRESS	46,00
Módulo N°9	DESARROLLO DE PORTAFOLIO DE UN PRODUCTO DIGITAL	18,00
Módulo N°10	DESARROLLO DE EMPLEABILIDAD EN LA INDUSTRIA DIGITAL	18,00
TOTAL DE HORAS		472,00

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



MÓDULO FORMATIVO N° 1		
Nombre	ORIENTACIÓN AL PERFIL Y METODOLOGÍA DEL CURSO	
N° de horas asociadas al módulo	18,00	
Código Módulo	MB00128	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Según el consignado en el plan formativo.	
Competencia del módulo	Identificar las competencias y habilidades del perfil profesional en junto al rol que poseen dentro del contexto de la industria TI así como la metodología de trabajo del curso para adquirirlas.	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Reconocer las características fundamentales del perfil profesional en cuestión, así como el contexto laboral en el cual se desenvuelve.	1.1 Identifica las competencias técnicas y personales valoradas por la industria TI. 1.2 Reconoce las competencias laborales del perfil en cuestión 1.3 Reconoce las características del contexto de la industria y del mercado laboral en el cual se desenvuelve el perfil en cuestión.	1. El perfil y la Industria TI: La industria TI. Características de la industria. Perfiles más comunes en la industria TI. Competencias técnicas valoradas por la industria TI. Habilidades personales valoradas por la industria TI. Metodologías y forma de trabajo del área. El mercado laboral TI. El perfil profesional asociado al curso (*). Qué es un perfil profesional. Competencias que posee el perfil. Habilidades que posee el perfil. Niveles de experiencia y seniority del perfil. Expectativas laborales del mercado actual para el perfil. Proyección laboral del perfil. (*) Considerar el perfil específico del plan formativo a ejecutar.
2. Distinguir la naturaleza de cada módulo que forma parte de la currícula así como las características del trabajo a desempeñar, productos obtenidos, y su contribución a la formación del perfil en cuestión.	2.1 Distingue las competencias que serán adquiridas a lo largo de cada módulo de la currícula. 2.2 Reconoce la naturaleza del trabajo que será realizado junto con las herramientas que serán utilizadas a lo largo de cada módulo. 2.3 Reconoce la importancia de un portafolio de producto así como sus características para la formación de una identidad profesional. 2.4 Identifica los productos que serán obtenidos en cada módulo así como su contribución al portafolio de producto.	2. Currícula del curso: La currícula del presente curso. Módulos y competencias a formar a lo largo de la currícula. Herramientas a utilizar durante el curso. Características del trabajo técnico a realizar en cada módulo. Productos obtenidos en cada módulo. El portafolio de producto. Qué es un portafolio de producto. Importancia de un portafolio de producto en la identidad profesional. Contribución de cada módulo del curso al portafolio.
3. Comprender la metodología de trabajo a lo largo del curso distinguiendo la importancia del trabajo en equipo y el trabajo ético para contribuir al logro de las competencias del plan formativo.	3.1 Describe los principales elementos de la metodología bootcamp para el desarrollo de competencias en el área tecnológica. 3.2 Reconoce las habilidades personales que se	3. Metodología del curso: Metodología de enseñanza-aprendizaje. Qué es un bootcamp y cuál es su origen. El día a día en un bootcamp de programación. Miedos habituales. Claves para enfrentar un bootcamp de

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



	<p>requiere formar a lo largo de un bootcamp para contribuir al logro de las competencias del plan formativo.</p> <p>3.3 Maneja herramientas de gestión de proceso de aprendizaje, de coordinación y de trabajo en equipo para contribuir al logro de las competencias del plan formativo.</p>	<p>programación. Exigencia y dedicación requerida. Importancia de la disciplina y perseverancia. Herramientas a utilizar a lo largo del bootcamp. Herramientas de gestión del proceso de aprendizaje (LMS). Herramientas de coordinación y trabajo colaborativo. Herramientas propias de la competencia técnica. Habilidades utilizadas a lo largo del bootcamp. Trabajo en Equipo: Importancia del trabajo en equipo, Buenas prácticas para el trabajo en equipo, Organización de los equipos, Autoaprendizaje, Importancia del autoaprendizaje, Cómo ir desarrollando la capacidad de autoaprendizaje. Tolerancia a la frustración: Qué es la tolerancia a la frustración, Relación con la programación, Técnicas para gestionar la tolerancia a la frustración. Comportamiento ético: Importancia del comportamiento ético en la industria TI. Código de ética del curso.</p>
PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
<p>* Formación profesional en el área de las ingenierías, para el caso de los planes formativos front-end, fullstack, mobile, o bien como diseñador para el caso del plan UX/UI. En todos los casos anteriores, con título.</p> <p>* Experiencia laboral de mínimo 3 años desempeñando funciones de laborales ya sea como desarrollador front-end, fullstack, mobile, o diseñador UX/UI, dependiendo del plan formativo, o bien gerenciando áreas en estas materias.</p> <p>* Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>* Formación académica como técnico de nivel superior en el área informática para el caso de los planes formativos front-end, fullstack, mobile, o bien como diseñador para el caso del plan UX/UI. En todos los casos anteriores, con título.</p> <p>* Experiencia laboral de mínimo 3 años desempeñando funciones de laborales ya sea como desarrollador front-end, fullstack, mobile, o diseñador UX/UI, dependiendo del plan formativo, o bien gerenciando áreas en estas materias.</p> <p>* Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>* Experiencia laboral de mínimo 6 años desempeñando funciones de laborales ya sea como desarrollador front-end, fullstack, mobile, o diseñador UX/UI, dependiendo del plan formativo, o bien gerenciando áreas en estas materias.</p> <p>* Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		
Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p> <p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p> <p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

<p>data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.</p>	<p>evidencias de actividades realizadas por los participantes.</p>	<p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del curso</p>
---	--	---

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



MÓDULO FORMATIVO N° 2		
Nombre	FUNDAMENTOS DE DESARROLLO FRONT-END	
N° de horas asociadas al módulo	72,00	
Código Módulo	MB00129	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Según el consignado en el plan formativo.	
Competencia del módulo	Implementar páginas web básicas responsivas utilizando HTML, CSS y JavaScript de acuerdo a los requerimientos y acorde a las buenas prácticas de la industria.	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Reconocer los principales aspectos relacionados con el desarrollo web distinguiendo el rol y elementos fundamentales de Front-End.	1.1 Reconoce el concepto de desarrollo web distinguiendo entre front-end, back-end y fullstack. 1.2 Describe las características del lenguaje html para el desarrollo web. 1.3 Reconoce el rol de html, css y javascript para el desarrollo de aplicaciones web front-end. 1.4 Identifica las herramientas requeridas para el desarrollo de aplicaciones front-end.	1. Aspectos principales del desarrollo en web: El desarrollo Web. Qué se entiende por desarrollo web. Diferencias entre Front-End, Back-End y Fullstack. Qué es el lenguaje de marcación de hipertexto (HTML). El rol del Navegador. Qué es la W3C. Evolución del html hacia el html5. La triada html, css y javascript: contenido, presentación y comportamiento. El Entorno de Desarrollo. Descarga del editor Visual Studio Code e Instalación. Utilizar el potencial de un editor de texto para el desarrollo. Conociendo el inspector de elementos en un navegador.
2. Utilizar el lenguaje de etiquetas html5 para la estructuración del contenido de una página web básica acorde a un requerimiento entregado.	2.1 Identifica la estructura básica de un documento html y los elementos que la componen. 2.2 Utiliza etiquetas de estructura del cuerpo para la organización de los contenidos en una página html. 2.3 Utiliza etiquetas para la estructuración de un formulario de captura de datos para dar solución a un problema. 2.4 Utiliza assets y rutas relativas para la estructuración de un documento HTML que da solución a un problema planteado. 2.5 Implementa un prototipo de contenido navegable utilizando etiquetas html para dar solución a un requerimiento.	2. El lenguaje HTML: El Lenguaje HTML. Características de html5. Elementos, etiquetas y atributos. Estructura básica de un documento: html, head, body, meta, title, link. Estructura del cuerpo: header, nav, section, aside, footer. Encabezados: H1...H7. Enlaces o hipervínculos. Imágenes. Listas y listas anidadas. Tablas. Formularios: form, input, select, radio, button, submit. Elemento DIV.
3. Aplicar hojas de estilo CSS básicas distinguiendo elementos de responsividad para personalizar la presentación de un documento HTML acorde a un	3.1 Reconoce los principios y usos de las hojas de estilo CSS para el manejo de los aspectos visuales básicos de un documento HTML.	3. Aplicación de estilos y responsividad: Manejando Hojas de Estilo. Qué es CSS, fundamentos y utilidad. CSS y HTML. Estilos en línea, embebidos, archivos

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



requerimiento entregado.	<p>3.2 Utiliza rutas absolutas y relativas para el manejo de assets e imágenes en la incorporación de hojas de estilo al documento HTML</p> <p>3.3 Codifica un documento HTML utilizando la sintaxis y reglas de estilos CSS para modificar aspectos visuales y resolver un problema planteado acorde a las buenas prácticas de la industria</p> <p>3.4 Identifica los conceptos claves de la responsividad de un reconociendo los mecanismos para implementarla en un documento html</p> <p>3.5 Utiliza las herramientas para desarrolladores provista por el navegador para la inspección de los estilos aplicados en el documento.</p>	<p>externos. Referencias y selectores, por clase, por id. El modelo de cajas. Estilos más utilizados (fuentes, líneas, cajas, etc...). Buenas prácticas al construir una hoja de estilos. Manejo de assets e imágenes. Conociendo rutas absolutas y relativas. Orden jerárquico de aplicación de reglas CSS y el peso asociado a las reglas. Inspeccionando estilos con las herramientas para desarrolladores en el navegador Responsividad. El concepto de Responsividad. Tipos de dispositivos y orientaciones. El concepto Mobile First. Utilización de Media Query. Cómo probar los distintos dispositivos.</p>
4. Implementar un sitio web básico responsivo utilizando framework Bootstrap para organizar la presentación de un documento html.	<p>4.1 Describe las características y beneficios de utilización del framework Bootstrap para el manejo de estilos en una página web.</p> <p>4.2 Reconoce los elementos y estilos principales de Bootstrap para el manejo de estilos en una página web.</p> <p>4.3 Utiliza estilos disponibles en el framework Bootstrap para organizar los elementos visuales en un documento HTML y resolver el problema planteado.</p>	<p>4. El Framework Bootstrap: El Framework Bootstrap. Qué es Bootstrap. Beneficios de su utilización. Dónde obtenerlo y cómo incorporarlo a un proyecto html. Elementos y estilos básicos de bootstrap: Containers, Grillas, Tablas, Jumbotron, Alertas, Botones, Imágenes, Navbars, Forms.</p>
5. Utilizar código Javascript para la personalización de eventos sencillos dentro de un documento html dando solución al problema planteado.	<p>5.1 Reconoce los elementos fundamentales y el rol del lenguaje JavaScript dentro de un documento html.</p> <p>5.2 Utiliza selectores básicos para la referenciación y obtención de textos y valores de elementos del DOM.</p> <p>5.3 Utiliza eventos básicos onClick y onChange para definir un comportamiento en elementos de un formulario.</p> <p>5.4 Utiliza variables, expresiones aritméticas y sentencias condicionales para la programación de una rutina simple en Javascript.</p> <p>5.5 Codifica funciones para la realización de operaciones sencillas dentro de un documento html.</p> <p>5.6 Utiliza la consola JavaScript para la depuración del código en la resolución de un problema.</p>	<p>5. Bases del lenguaje Javascript: Bases del Lenguaje JavaScript. Breve historia de JavaScript. Relevancia de Javascript. Qué puede y no puede hacer en el contexto de un navegador. Cómo incorporar código Javascript en un documento html. Selectores básicos: getElementById. Obtención y manipulación de valores y textos de los elementos del DOM. Eventos básicos: onClick y onChange. Variables. Expresiones aritméticas. Sentencias. condicionales. Funciones. Cómo ejecutar código Javascript en la consola. Depurando el código Javascript con la consola.</p>
6. Utilizar biblioteca JQuery para la incorporación de elementos dinámicos en un documento html mediante el manejo básico de objetos y eventos del DOM.	<p>6.1 Describe los aspectos fundamentales y rol de una biblioteca javascript en el desarrollo front-end.</p> <p>6.2 Utiliza la biblioteca JQuery para la selección y manipulación de elementos básicos del DOM que permiten resolver el problema planteado.</p> <p>6.3 Utiliza la biblioteca JQuery para el manejo de eventos básicos dando solución a un problema</p>	<p>6. Conociendo la librería JQuery: JQuery básico: La biblioteca JQuery. Por qué y cuándo usarla. Obtener JQuery. Incluir y usarlo en un sitio. Bootstrap y JQuery. Qué es el DOM. Manipulación de elementos del DOM con JQuery. Eventos, tipos de evento, cómo interactuar con ellos. Plugins Bootstrap-jQuery. ¿Qué es y cuándo usar un plugin?. Incorporar un plugin bootstrap-jQuery a</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

	planteado. 6.4 Utiliza plugins de bootstrap-JQuery para implementar componentes de terceros en una página web.	la página web. Ejemplos de plugins bootstrap-jQuery más comunes.
7. Gestionar el código fuente de un proyecto utilizando GitHub para mantener un repositorio de código remoto seguro y permitir trabajo concurrente.	7.1 Describe el rol y las principales características de un sistema de control de versiones para la gestión del código fuente. 7.2 Gestiona el código fuente en un repositorio local, realizando consolidación y restauración de archivos utilizando GIT. 7.3 Gestiona el código fuente mediante ramas y su posterior unión resolviendo conflictos existentes utilizando GIT. 7.4 Gestiona el código fuente utilizando repositorios locales y remotos, sincronizando y resolviendo conflictos existentes utilizando GIT. 7.5 Gestiona el trabajo colectivo en un repositorio remoto administrando Pull Requests utilizando GIT.	7. Fundamentos de GIT y GitHub: Fundamentos de GIT. Necesidad de un repositorio de código fuente. Instalación, configuración y comandos básicos. Commits y restauración de archivos. Cambios de nombres. Ignorando archivos. Ramas, uniones, conflictos y tags. Stash y Rebase. Fundamentos de GitHub. Repositorios remotos, Push y Pull. Fetch v/s Pull. Clonando un repositorio. Documentando un proyecto con Markdown. Administrando Pull Request. Flujos de trabajo con GitHub.
PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
* Formación profesional en el área de la ingeniería informática, electrónica, industrial, con título. * Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo front-end web, demostrable. * Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.	* Formación académica como Técnico de Nivel Superior en el área informática, con título. * Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo front-end web, demostrable. * Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.	* Experiencia laboral de mínimo 6 años desempeñando funciones de desarrollo front-end web, demostrable. * Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		
Infraestructura	Equipos y herramientas	Materiales e insumos
*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador. *Conexiones para utilizar medios didácticos tales como data y salida a internet. *Sistema de iluminación y ventilación adecuados *Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.	*Notebook o PC para facilitador. *Proyector multimedia. *Telón. *Pizarrón. *Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes. *Equipo computacional, uno para cada participante, al menos Pentium Core 5, 8 GB RAM y 500 GB HD, y	*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta). *Material de apoyo y guía sobre el sector. *Plumones para pizarrón. *Libro de clases. *Pautas de evaluación por actividad. *Manual del participante que contemple todos los contenidos especificados para este módulo.

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

*Para actividades E-Learning debe contar con plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.	conexión a internet, con los siguientes programas: Visual Studio Code; Sublime Text; Spring STS 4 ó superior.	
---	---	--

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



MÓDULO FORMATIVO N° 3		
Nombre	FUNDAMENTOS DE PROGRAMACIÓN EN JAVASCRIPT	
N° de horas asociadas al módulo	60,00	
Código Módulo	MB00130	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Según el consignado en el plan formativo.	
Competencia del módulo	Codificar piezas de software de baja/mediana complejidad utilizando lenguaje Javascript para resolver problemáticas comunes de acuerdo a las necesidades de la industria.	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Reconocer las características fundamentales del lenguaje Javascript para el desarrollo web	1.1 Reconoce conceptos básicos del lenguaje Javascript para el desarrollo web 1.2 Distingue características propias del lenguaje Javascript versus otros lenguajes para el desarrollo web 1.3 Reconoce el entorno de ejecución y las herramientas basadas en Javascript para el desarrollo de aplicaciones web	1. Introducción al lenguaje Javascript: El Navegador web y entorno de ejecución. Herramientas del browser. Qué puede y no puede hacer Javascript en el Navegador. La consola de comandos Javascript. Herramientas para desarrollo Javascript.
2. Utilizar variables simples y sentencias condicionales para el control del flujo de un algoritmo que resuelve un problema simple acorde al lenguaje Javascript	2.1 Utiliza instrucciones condicionales en un algoritmo para dar solución a un problema acorde al lenguaje Javascript 2.2 Distingue variables con tipos de dato simple y complejos, así como scope de variables acorde al lenguaje Javascript 2.2 Codifica una rutina simple en Javascript a partir de un diagrama de flujo para dar solución a un problema 2.3 Esboza un diagrama de flujo utilizando la notación requerida para representar un algoritmo condicional 2.4 Utiliza la consola de Javascript y el inspector de elementos del navegador para verificar el algoritmo	2. Variables, expresiones y sentencias condicionales Variables: Qué es una variable. Consejos para nombrar una variable. Declaración de una variable. Inicialización de una variable. Variables constantes. Tipos de variable simples: String, number, boolean. Tipos de variable complejas: arreglos, objetos. Scope de variables. Operaciones y expresiones. Operadores aritméticos. Precedencia de operadores. Operadores de incremento y decremento. Operadores de comparación Cadenas de caracteres. Creación de una cadena de caracteres, uso de comillas. Concatenación. Sentencias Condicionales. Expresiones y operaciones lógicas. Manejando flujo con if y else. Implementando un código a partir de un diagrama de flujo. Manejando condiciones de borde.
3. Utilizar estructuras de tipo arreglo y sentencias iterativas para el control del flujo de un algoritmo que resuelve un problema simple acorde al lenguaje Javascript	3.1 Identifica las características y utilidad de una estructura de tipo arreglo para la elaboración de un algoritmo 3.2 Utiliza operaciones de creación, agregación, modificación y eliminación de elementos de un arreglo	3. Arreglos y ciclos Arreglos: Qué es un arreglo y cuándo lo necesitamos. Crear Arreglos. Acceder a un elemento en un arreglo. Contar los elementos de un arreglo. Iterar sobre los elementos de un arreglo. Insertar y borrar elementos

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



	<p>acorde al lenguaje Javascript para resolver un problema</p> <p>3. 3 Distingue el scope de las variables que se utilizan en bloques cíclicos acorde al lenguaje Javascript</p> <p>3.4 Utiliza ciclos anidados y condiciones de salida para resolver un problema</p> <p>3.5 Utiliza combinaciones de ciclos con instrucciones if/else para resolver un problema</p> <p>3.6 Utiliza estilos y convenciones de programación tales como indentación y estructura de código acorde al lenguaje Javascript y las buenas prácticas</p>	<p>dentro de un arreglo. Álgebra con arreglos (Operaciones de unión, intersección, concatenación y diferencia). Matrices y arreglos asociativos</p> <p>Ciclos iterativos. Para qué sirven los ciclos. Creando un ciclo con while. Asignación con +=. Sumatorias. Scope de variables en un ciclo. Instrucción do/while. Instrucción For. Ciclos anidados. Combinación de ciclos con instrucciones if/else. Estilos, convenciones y buenas prácticas de codificación. El concepto de código limpio. Por qué seguir una guía y convenciones. Convenciones y buenas prácticas.</p>
4. Codificar un programa utilizando funciones para la reutilización de código acorde al lenguaje Javascript	<p>4.1 Identifica las características y utilidad de programar funciones en un programa</p> <p>4.2 Distingue el alcance de una variable dentro y fuera de una función acorde al lenguaje Javascript</p> <p>4.3 Codifica funciones personalizadas definiendo parámetros de entrada y retorno para resolver un problema acorde al lenguaje Javascript</p> <p>4.5 Codifica un programa invocando funciones personalizadas para resolver un problema acorde al lenguaje Javascript</p>	<p>4. Funciones en Javascript: Qué es una función y para qué sirve. Definir funciones. Paso de parámetros en una función. Retorno de una función. Variables locales y variables globales. Invocación de una función. Alcance de las variables locales. El problema de las variables globales. Crear una función anidada.</p>
5. Utilizar objetos preconstruidos para la codificación de un algoritmo que resuelve un problema acorde al lenguaje Javascript	<p>5.1 Identifica las características principales de los objetos en el lenguaje Javascript así como sus objetos preconstruidos más utilizados</p> <p>5.2 Utiliza el objeto Math para la construcción de un algoritmo que resuelve un problema</p> <p>5.3 Utiliza el objeto String para la construcción de un algoritmo que resuelve un problema</p>	<p>5. Conceptos básicos de Objetos en Javascript: Qué es un objeto. Elementos que contiene un objeto: propiedades y métodos. Notación de punto. Notación de corchetes. Métodos dentro de un objeto. Objetos preconstruidos en Javascript: El objeto Math, Qué es el objeto Math y para qué sirve, Algunas propiedades del objeto Math, Métodos usuales del objeto Math. El objeto String: Qué es el objeto String y para qué sirve, Creación de un objeto String versus uno primitivo, Métodos usuales del objeto String.</p>
PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
<p>*Formación profesional en el área de la ingeniería informática, electrónica, industrial, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web front-end con Javascript, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades,</p>	<p>*Formación académica como técnico de nivel superior en el área informática, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web front-end con Javascript, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades,</p>	<p>*Experiencia laboral de mínimo 6 años desempeñando funciones de desarrollo web front-end con Javascript, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.	institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.	
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		
Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p> <p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes.</p> <p>* Equipo computacional, uno para cada participante, al menos Pentium Core 5, 8 GB RAM y 500 GB HD, conexión a internet; Software instalado: Visual Studio Code Editor; Editor de código Atom; Google Chrome o Mozilla Firefox</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p> <p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p> <p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del curso.</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



MÓDULO FORMATIVO N° 4		
Nombre	PROGRAMACIÓN AVANZADA EN JAVASCRIPT (V2)	
N° de horas asociadas al módulo	48,00	
Código Módulo	MB00131	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Según el consignado en el plan formativo.	
Competencia del módulo	Codificar piezas de software de baja/mediana complejidad en lenguaje Javascript utilizando paradigmas de orientación a objetos, orientación a eventos y programación asíncrona para resolver problemáticas de acuerdo a las necesidades de la organización.	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Utilizar los conceptos fundamentales de la programación orientada a objetos acorde al lenguaje Javascript para resolver un problema simple.	1.1 Describe las características del paradigma de la programación orientada a objetos para la programación web. 1.2 Reconoce los pilares de la programación orientada a objetos para el desarrollo de una pieza de software. 1.3 Codifica un programa utilizando los mecanismos de creación de objetos con propiedades y métodos para resolver un problema simple de orientación a objetos. 1.4 Aplica la notación de objetos Javascript para la definición de un objeto o arreglo de objetos de mediana complejidad.	1. Orientación a Objetos en Javascript: Conceptos de la POO. Paradigmas de programación. Abstracción y Programación Orientada a Objetos. Clases, objetos e instancias. Elementos de una clase: propiedades y métodos. Pilares de la POO: herencia, polimorfismo, encapsulamiento y abstracción Javascript orientado a objetos. Creación de objetos literales. Creación de objetos mediante los constructores. La propiedad prototipo para definir métodos. La notación de objetos JavaScript. Qué es un JSON y para qué sirve. Estructura de un JSON. Arreglos como JSON. Herramientas online para construir y validar estructuras JSON.
2. Utilizar las nuevas funcionalidades de la especificación ES6+ para la implementación de un algoritmo Javascript que resuelve un problema planteado	2.1 Identifica las características esenciales de la especificación ES6 para la programación Javascript reconociendo el problema de la compatibilidad de los browsers. 2.2 Reconoce las diferencias entre el uso de Var, Let y Const para la asignación de una variable acorde al lenguaje Javascript ES6. 2.3 Utiliza la nueva nomenclatura para la definición de una clase, atributos y métodos acorde al lenguaje Javascript ES6. 2.4 Identifica las nuevas funcionalidades de la especificación ES6 referentes a ejecuciones asíncronas.	2. Características Javascript ES6: JavaScript especificación ES6+. Qué es ES6. Breve historia de ES6. Conocer compatibilidad de ES6 con navegadores actuales. Conocer tecnologías para facilitar la integración con navegadores. Webpack. Babel. Polyfills. Características nuevas ES6+. Variables. Diferencias entre Var, Let y Const. Funciones. Arrow functions, ¿Cuándo usarlas?. Parámetros por defecto. Strings. Interpolado de strings. Objetos. Destructuring de objetos o arrays. Operador Spread. Operador Rest. Asignación concisa de atributos. Clases. Definición. Herencia. Atributos. Módulos, exportar e importar. Sets y Maps. Iteradores y Generadores. Promesas. Async y Await.

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



3. Reconocer los elementos fundamentales del Domain Object Model y los mecanismos para la manipulación de elementos en un documento html.	3.1 Identifica los elementos fundamentales que componen la jerarquía del Document Object Model. 3.2 Utiliza instrucciones Javascript para la manipulación de un objeto utilizando la jerarquía del DOM. 3.3 Explica el rol de los eventos dentro del DOM reconociendo los más frecuentes. 3.4 Utiliza instrucciones Javascript para la definición de un comportamiento simple ante un evento en el DOM determinado.	3. Eventos y manipulación del DOM: Document Object Model. Qué es el DOM. Jerarquía de elementos del DOM. Manipulación del DOM. Seleccionar un elemento. Modificar un elemento. Agregar un elemento. Modificar un estilo. Eventos: Qué es un evento, Algunos eventos frecuentes: click, focus, scroll, load.
4. Utilizar elementos de programación asíncrona para resolver un problema simple distinguiendo los diversos mecanismos para su implementación acorde al lenguaje Javascript.	4.1 Explica el concepto de programación asíncrona reconociendo el problema que resuelve y los mecanismos disponibles en Javascript. 4.2 Distingue las diferencias en la utilización de los distintos mecanismos de programación asíncrona (callbacks, promises, async/await) reconociendo sus ventajas y desventajas. 4.3 Utiliza instrucciones de ejecución asíncrona (callbacks, promises, async/await) para resolver un problema simple de asincronía acorde al lenguaje Javascript. CE 4.4 Utiliza generación y captura de errores personalizados para resolver un problema simple de programación asíncrona acorde al lenguaje Javascript.	4. Javascript Asíncrono: Qué es la programación asíncrona. Qué es un Thread. Código asíncrono v/s código bloqueante. Callbacks asíncronos. Qué es una función asíncrona. Qué es un callback. Creando una función callback. Ejemplo con SetTimeout. Paso de parámetros en funciones callback. Invocando una llamada callback. Promises. Qué son las Promesas. Ventajas de las Promesas por sobre los Callbacks. Los bloques .then() y .catch(). Construyendo una promesa. Resolviendo una promesa. Rechazando una promesa Async/Await. Creando funciones asíncronas con ASYNC/AWAIT. Capturar un error con Catch. Ventajas y desventajas
5. Utilizar el objeto XHR y la API Fetch para el consumo de una API externa y su procesamiento acorde al lenguaje Javascript.	5.1 Describe qué es una API y su importancia para el desarrollo web distinguiendo las APIs de browser y APIs de terceros. 5.2 Utiliza la API XHR para la obtención de información desde un servidor para resolver un problema simple acorde al lenguaje Javascript. 5.3 Utiliza la API Fetch para la obtención de información desde un servidor para resolver un problema simple acorde al lenguaje Javascript. 5.4 Codifica un programa que despliega datos de una respuesta JSON accediendo a elementos del DOM utilizando lenguaje Javascript para resolver un problema.	5. Consumo de APIs con Javascript: APIS en Javascript. Qué es una API y para qué sirve. APIs Javascript del lado del cliente: APIs del browser, APIs de terceros. Diferencia entre API del browser, API de terceros, librería Javascript, Framework Javascript. APIs del browser comunes. APIs de terceros populares. Obtención de data desde un servidor. Qué es Ajax y qué problema resuelve. Las APIs XHR y Fetch, características de cada una. Realizando requests con XHR. Realizando requests con Fetch. Cuándo usar una u otra APIs de terceros. Qué es una API de terceros. Ejemplos populares de APIs de terceros. Conectando la API con una api-key. Solicitando data a la API. Procesamiento de la respuesta.
PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
*Formación profesional en el área de la ingeniería	*Formación académica como técnico de nivel superior	*Experiencia laboral de mínimo 6 años desempeñando

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



<p>informática, electrónica, industrial, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web front-end con Javascript, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>en el área informática, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web front-end con Javascript, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>funciones de desarrollo web front-end con Javascript, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		
Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p> <p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes.</p> <p>*Equipo computacional, uno para cada participante, al menos Pentium Core 5, 8 GB RAM y 500 GB HD, conexión a internet, con el siguiente software instalado: Visual Studio Code Editor; Editor de código Atom; Google Chrome o Mozilla Firefox; Postman.</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p> <p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p> <p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del curso</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

MÓDULO FORMATIVO N° 5		
Nombre	FUNDAMENTOS DE BASES DE DATOS RELACIONALES	
N° de horas asociadas al módulo	56,00	
Código Módulo	MB00134	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Según el consignado en el plan formativo.	
Competencia del módulo	Operar una base de datos relacional utilizando el lenguaje SQL para la obtención, manipulación y definición de datos dando solución a un problema de almacenamiento de información	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Identificar las características, rol y elementos fundamentales de una base de datos relacional para la gestión de la información en una organización	1.1 Explica el rol de una base de datos en la organización 1.2 Identifica las características de una base de datos para la gestión de la información 1.3 Reconoce los principales objetos que componen una base de datos para la gestión de la información	1. Las Bases de Datos Relacionales: Las Bases de Datos Relacionales. El rol de las bases de dato relacionales en la organización. Características de un RDBMS. Alternativas de BD más utilizadas en la industria. Conociendo las herramientas para consultar una base de datos. Instalando la base de datos y sus herramientas utilitarias. Creando una conexión a la base de datos. Los principales objetos de una base de datos.
2. Utilizar lenguaje estructurado de consultas SQL para la obtención de información que satisface los requerimientos planteados a partir de un modelo de datos dado	2.1 Reconoce los elementos fundamentales del lenguaje SQL para la obtención de datos de una base de datos relacional 2.2 Utiliza sentencias SQL con condiciones de selección para resolver un problema planteado de selección condicional 2.3 Utiliza sentencias SQL que requieren la consulta a varias tablas relacionadas a partir de un modelo de datos dado para resolver un problema planteado de selección 2.4 Utiliza sentencias SQL con funciones de agrupación para resolver un problema planteado que requiere la agrupación de datos	2. Consultas a una o varias tablas: Consultando información de una tabla. El Lenguaje Estructurado de Consultas SQL. Recuperando información de una tabla. Consultas utilizando la llave primaria. Consultas utilizando condiciones de selección. Utilización de funciones en las consultas. Consultas de selección con funciones de agrupación. Consultando información relacionada en varias tablas. Qué es un modelo de datos y cómo leerlo. Consultas de selección con tablas relacionadas. Integridad referencial. Querys anidadas. Querys con distintos tipos de JOIN (INNER, LEFT, OUTER).
3. Utilizar lenguaje de manipulación de datos DML para la modificación de los datos existentes en una base de datos dando solución a un problema planteado	3.1 Reconoce la sintaxis básica para la construcción de sentencias DML que resuelven un requerimiento de manipulación de datos 3.2 Utiliza sentencias de ingreso, actualización y borrado de registros en una tabla utilizando lenguaje DML de acuerdo a las condiciones solicitadas	3. Sentencias para la manipulación de datos y transaccionalidad: Sentencias para la manipulación de datos. Data Manipulation Lenguaje (DML). Actualizando la información de una tabla. Borrando información de una tabla. Ingresando información a una tabla. Utilización de secuencias para asignar identificadores.

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



	<p>3.3 Utiliza sentencias de ingreso, actualización y borrado de registros utilizando lenguaje DML para manipular la información de tablas con integridad referencial de acuerdo a un modelo de datos existente</p> <p>3.4 Reconoce el concepto de transaccionalidad y su importancia para mantener la consistencia de la información en una base de datos</p>	<p>Insertar, actualizar y borrar datos con integridad referencial. Restricciones en una tabla. Transaccionalidad en las operaciones. Qué es una transacción y por qué son importantes. Propiedades de las transacciones: atomicidad, consistencia, aislamiento, durabilidad. Confirmación de una transacción. Vuelta atrás de una transacción. Modo autocommit.</p>
4. Implementar estructuras de dato relacionales utilizando lenguaje de definición de datos DDL a partir de un modelo de datos para la creación y mantención de las definiciones de los objetos de una base de datos	<p>4.1 Reconoce la sintaxis básica para la creación de expresiones DDL que resuelven un requerimiento de definición de datos</p> <p>4.2 Construye sentencias de creación de una tabla utilizando DDL y definiendo campos, tipos de dato, nulidad, llaves primarias y foráneas de acuerdo a un modelo de datos existente para satisfacer un requerimiento</p> <p>4.3 Construye sentencias utilizando DDL para la modificación de los atributos de una tabla de acuerdo a los requerimientos planteados</p>	<p>4. Sentencias para la definición de tablas: Sentencias para la definición de tablas. El lenguaje de definición de datos DDL. Creación de una tabla: Definición de campos, Tipos de datos, La restricción de nulidad, Definición de la llave primaria, Definición de llaves foráneas, Creando un modelo de datos con integridad referencial. Modificación de una tabla: Modificar un campo en una tabla, Modificar una condición de nulidad. Eliminación de una tabla. Truncado de una tabla</p>
5. Elaborar un modelo de datos de acuerdo a los estándares de modelamiento para resolver un problema de baja complejidad	<p>5.1 Describe las principales características de un modelo de Entidad-Relación para el modelamiento de un problema de datos</p> <p>5.2 Elabora un modelo de Entidad-Relación identificando entidades, identificadores y relaciones entre entidades para representar un problema dado</p> <p>5.3 Reconoce las reglas de transformación y de normalización (3F) de un modelo relacional</p> <p>5.4 Elabora un diccionario de datos detallando un modelo relacional que resuelve un problema dado</p>	<p>5. El modelo Entidad-Relación: El modelo Entidad-Relación. El proceso de abstracción. El modelo conceptual de Entidad-Relación. Identificación de entidades. Definición de atributos e identificadores únicos. Tipos de relación entre entidades. Entidades débiles y fuertes. El modelo Relacional. El modelo relacional y sus diferencias con el modelo conceptual. Reglas de transformación. Asignando tipos de datos y restricciones al modelo. Normalización y Desnormalización de datos. El diccionario de datos.</p>
PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
<p>*Formación profesional en el área de la ingeniería informática, electrónica, industrial, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo de aplicaciones con bases de dato, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Formación académica como técnico de nivel superior en el área informática, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo de aplicaciones con bases de dato, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Experiencia laboral de mínimo 6 años desempeñando funciones de desarrollo de aplicaciones con bases de dato, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p> <p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes.</p> <p>*Equipo computacional, uno para cada participante, al menos Pentium Core 5, 8 GB RAM y 500 GB HD, y conexión a internet; Software Motor BD (puede ser en un servidor central, en la nube o bien local). Opciones: Oracle 11g, MySQL 7, Postgres 9. O bien, versiones superiores; Software cliente para realizar consultas a la BD. Opciones: Oracle Developer, PgAdmin4, PhpMyAdmin, o equivalentes.</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p> <p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p> <p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del curso</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



MÓDULO FORMATIVO N° 6		
Nombre	DESARROLLO DE APLICACIONES WEB NODE EXPRESS	
N° de horas asociadas al módulo	72,00	
Código Módulo	MA03005	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Licencia de Enseñanza Media completa; Manejo computacional a nivel usuario; Aprobación de test de selección en etapa I y II, consistente en prueba en línea que mide habilidades de raciocinio lógico y comprensión verbal, más un test de aptitud. **Se exceptúan de este requisito, a los postulantes al programa Talento Digital para Chile, que estén cursando cuarto año medio en liceos científico-humanista o en el caso de alumnos de liceos técnico profesional que estén cursando cuarto medio o desarrollando su práctica laboral."	
Competencia del módulo	Desarrollar aplicaciones web utilizando el entorno multiplataforma Node Express para dar solución a los requerimientos de la organización	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Describir las características fundamentales del entorno Node.js y su utilidad para el desarrollo de aplicaciones web.	1.1 Reconoce las características básicas del entorno Node Express identificando sus componentes para el desarrollo de aplicaciones web. 1.2 Describe el uso del entorno Node Express y las problemáticas que son posibles de resolver para el desarrollo de aplicaciones web. 1.3 Reconoce las ventajas y potencialidades de Express para el desarrollo de aplicaciones empresariales bajo el entorno Node.js. 1.4 Distingue la aplicación de tareas con Node puro versus Express integrado para el desarrollo de aplicaciones en el entorno Node.js. 1.5 Reconoce el procedimiento de Incorporación de Express como integrador de un proyecto Web acorde al entorno Node.js.	1. Conociendo Node y Express: Qué es Node. Código Javascript en el servidor. El motor V8 de Google. Peticiones asíncronas y threads. Delegando tareas a procesos externos. Todo es Callback. Un único hilo de ejecución Qué podemos hacer con Node. Ejecución de tareas simultáneas. Disponibilizar servicios API REST. Conectar a Bases de Datos. Consumir API REST. Acceder al file system. Utilizar Socket. Desarrollar aplicaciones Web completas. Testear aplicaciones. Otros. El framework Express. Qué es Express. Por qué usar Express. Express v/s Node. Express y la infraestructura Web. Ventajas: Velocidad de desarrollo, Estructura minimalista, Estructura Flexible. Ruteo declarativo. Desarrollo de API's. Qué es un middleware. Patrones básicos de middleware. Motor de plantillas. Renderización dinámica. Depuración de código.
2. Aplicar procedimiento de Instalación y configuración del ambiente Node Express básico para el desarrollo de una aplicación web.	2.1 Describe las características fundamentales del entorno Node para el desarrollo de aplicaciones web. 2.2 Aplica procedimiento de Instalación del entorno Node Express para la creación de un programa básico. 2.3 Aplica procedimiento de creación de un programa Node Express simple para verificar la instalación del	2. Instalación y configuración de Node: Instalando Node y Express desde la consola de comandos. Instalando Visual Studio Code. Levantar la consola de comandos. Crear un proyecto Node. Crear el primer programa Node. Crear el primer "Hola Mundo". Ejecutar por línea de comandos. Revisar el estado de la consola. Mostrar

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



	entorno.	resultados a la consola. Probar instrucciones blocking v/s non-blocking. Ciclo de vida de un proceso Nod. Una breve mirada a los paquetes más usados. Nodemon. Yargs. Moment. Socket.io. Mocha. Underscore. Morgan. Instalación de nodemon.
3. Utilizar el gestor NPM para la administración de dependencia y paquetes reconociendo el rol que tiene dentro del ciclo de vida de un proyecto Node.js.	3.1 Describe el rol del gestor de paquetes NPM en el ciclo de vida de un proyecto Node. 3.2 Utiliza el gestor NPM para la instalación de paquetes en el entorno Node configurando sus versiones. 3.3 Utiliza un módulo dentro de un programa Node mediante sentencias de importación para el uso de sus funcionalidades.	3. Gestión de paquetes en Node: El gestor de paquetes NPM. Qué es npm. Para qué sirve npm. Instalando paquetes básicos. Manejo de versión de paquetes. Desinstalando paquetes. Argumentos de la línea de comandos npm. Comandos básicos npm: Init. Install. Uninstall. Update. List. Outdated. La carpeta node_modules. Uso de paquetes en Node. Requiriendo paquetes en Node. Importar archivos de nuestro proyecto. Exportando módulos. Entendiendo el archivo package.json.
4. Implementar un servidor web de contenidos estáticos y dinámicos utilizando motores de plantillas acorde al entorno Node Express para dar solución a un problema.	4.1 Describe las características y el rol de un motor de plantillas para el despliegue dinámico de contenidos. 4.2 Implementa un servidor de contenidos estáticos utilizando el entorno Express y Node.js. 4.3 Utiliza motor de plantillas Handlebars para el despliegue de contenido dinámico. 4.4 Implementa una vista dinámica utilizando renderización de objetos, variables, páginas parciales y helpers para resolver un problema.	4. Sirviendo contenido web: Configuración del servidor. Configuración de rutas. Cómo servir contenido estático. Carpeta para contenido público. Cómo usar páginas parciales. Cómo desplegar datos variables en las páginas. Qué es un motor de plantillas. Alternativas de motores de plantillas: Handlebars, Ejs, Pug. Instalando Handlebars (HBS). Configuración paquete HBS. Renderizar objetos y variables en la Vista. Páginas parciales. Utilización de helpers.
5. Implementar la persistencia de objetos en una aplicación web utilizando archivos de texto plano para resolver un problema acorde al entorno Node.js.	5.1 Reconoce los módulos requeridos para trabajar con archivos en un proyecto acorde al entorno Node.js. 5.2 Utiliza instrucciones de lectura y escritura de objetos JSON en archivos de texto para dar solución a un problema acorde al entorno Node.js.	5. Persistencia en archivos planos: Instalar paquetes necesarios. Almacenando objetos JSON en archivos. Leer información de un archivo y cargarlo en objetos JSON. Modificar información de un archivo. Eliminar información de un archivo. Separar las funcionalidades en archivos independientes para optimizar nuestro código.
6. Aplicar procedimiento de puesta en ejecución de una aplicación Node.js implementando mecanismos para la detección de errores durante la ejecución.	6.1 Aplica procedimiento para levantar una aplicación Node utilizando paso de parámetros por línea de comandos. 6.2 Ejecuta procedimiento para bajar una aplicación Node utilizando la línea de comandos. 6.3 Interpreta los errores comunes desde la salida a consola al levantar y ejecutar una aplicación Node.	6. Ejecución de un aplicativo Node: Invocación con paso de parámetros. Paso de parámetros al iniciar nuestro programa. Validación de paso de parámetros. Uso del paquete YARGS. Validaciones automáticas con YARGS. Bajando una aplicación. Funciones globales en Node. Salida de la consola. Entendiendo la salida de la consola comandos. Coloreando la salida. Errores legibles en la línea de comandos. Levantar con Nodemon. Errores en la salida de la consola de comandos. Errores comunes al levantar una aplicación Node. Los mensajes de console.log(). Resolviendo errores comunes.

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
<p>*Formación profesional en el área de la ingeniería informática, electrónica, industrial, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web con Javascript y el entorno Node.js., demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Formación académica como técnico de nivel superior en el área informática, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web con Javascript y el entorno Node.js., demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Experiencia laboral de mínimo 6 años desempeñando funciones de desarrollo web con Javascript y el entorno Node.js., demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		
Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p> <p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes.</p> <p>*Equipo computacional, uno para cada participante, al menos Pentium Core 5, 8 GB RAM y 500 GB HD, conexión a internet, con el siguiente software instalado: Node 10.X o superior compatible; Visual Studio Code Editor 1.3 o superior; Google Chrome 70.0+ o Mozilla Firefox 64.0+; Postman 7.9 o superior.</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p> <p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p> <p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del curso</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

MÓDULO FORMATIVO N° 7		
Nombre	ACCESO A DATOS EN APLICACIONES NODE	
N° de horas asociadas al módulo	64,00	
Código Módulo	MA03006	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Licencia de Enseñanza Media completa; Manejo computacional a nivel usuario; Aprobación de test de selección en etapa I y II, consistente en prueba en línea que mide habilidades de raciocinio lógico y comprensión verbal, más un test de aptitud. **Se exceptúan de este requisito, a los postulantes al programa Talento Digital para Chile, que estén cursando cuarto año medio en liceos científico-humanista o en el caso de alumnos de liceos técnico profesional que estén cursando cuarto medio o desarrollando su práctica laboral."	
Competencia del módulo	Desarrollar una aplicación que ejecute operaciones de consulta y manipulación de datos utilizando el entorno Node.js de acuerdo a las buenas prácticas de la industria	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Implementar la conexión a una base de datos PostgreSQL en una aplicación Node utilizando buenas prácticas.	1.1 Reconoce las dependencias y paquetes requeridos para la implementación de una conexión a una base de datos PostgreSQL en el entorno Node. 1.2 Distingue las características, ventajas y desventajas de una conexión de tipo pooling en comparación a una simple. 1.3 Utiliza sentencias de conexión y desconexión a una base de datos PostgreSQL utilizando el entorno Node.js. 1.4 Utiliza sentencias para la captura y procesamiento de errores de conexión y estados de base de datos utilizando el entorno Node.js.	1. Conexión a una base de datos: Instalación de paquetes necesarios. El paquete pg de Node para PostgreSQL. Configurando el paquete pg. Incluyendo el paquete pg en un programa Node Conectando a la base de datos. Definiendo credenciales de acceso. El cliente de conexión. Conectando a la base de datos. Pooling de conexiones. Qué es pooling. Cuándo usar pooling. Ventajas y desventajas de usar pooling. El string de conexión URI. Desconectar de la base de datos. Capturando errores y mensajes. Errores comunes de conexión. Atrapando errores al conectar. Resolviendo errores de conexión. Formas correctas de conectar a una base de datos.
2. Programar instrucciones para la obtención de información desde una base de datos utilizando buenas prácticas acorde al entorno Node.js.	2.1 Utiliza sentencias para la consulta de datos mediante una conexión simple o un pool de conexiones para dar solución a un requerimiento. 2.2 Distingue los mecanismos para la implementación de consultas síncronas y asíncronas de acuerdo al entorno Node.js. 2.3 Identifica los riesgos de SQL Injection y las medidas para mitigarlo en una consulta de obtención de datos. 2.4 Programa instrucciones de obtención de datos utilizando Prepared Statements para mitigar el riesgo de	2. Obtención de información desde una base de datos: Consultas en base de datos. Consultas sencillas con la instrucción query(). Client.query(). Pool.query() Consultas Asíncronas. Consultas asíncronas con ASYNC/AWAIT. Consultas asíncronas con Callbacks. Consultas Parametrizadas. Queries con texto plano. Queries con texto parametrizado. Qué es SQL Injection. Validación parámetros en consultas. Objeto JSON como argumento de consultas. Prepared Statements. Row Mode. Types. Tipos de datos. Procesamiento de la

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



	<p>SQL Injection.</p> <p>2.5 Describe el concepto de cursor para la obtención de información en una base de datos acorde al entorno Node.js.</p> <p>2.6 Utiliza cursores para el procesamiento de una consulta de obtención de información acorde al entorno Node.js.</p>	<p>respuesta. El objeto result(). Iteración sobre filas y columnas. Captura de errores en consultas</p> <p>Manejo de cursores en Node. Qué es un cursor. Instalación de paquete. Constructor. Lectura e Iteración sobre cursores. Cierre de cursores.</p>
3. Programar instrucciones para la manipulación de información en una base de datos utilizando buenas prácticas acorde al entorno Node.js.	<p>3.1 Utiliza sentencias para la inserción, actualización y borrado de datos utilizando el entorno Node.js.</p> <p>3.2 Utiliza sentencias para la captura y manejo de errores durante las operación de manipulación acorde al entorno Node.js.</p>	<p>3. Modificación de datos en una base de datos: Inserción de datos. Instrucción INSERT y paso de parámetros. Obtener cantidad de registros insertados. Recuperación de último id insertado. Actualización de datos. Instrucción UPDATE y paso de parámetros. Obtener cantidad de registros actualizados. Borrado de datos. Instrucción DELETE y paso de parámetros. Obtener cantidad de registros eliminados. Capturando errores en manipulación de datos.</p>
4. Implementar operaciones transaccionales en una base de datos para mantener la consistencia de los datos utilizando el entorno Node.js.	<p>4.1 Describe conceptos básicos de transaccionalidad y sus beneficios para la implementación de un aplicativo que interactúa con una base de datos.</p> <p>4.2 Utiliza sentencias para el inicio, confirmación y vuelta atrás de una transacción dando solución a un problema.</p> <p>4.3 Utiliza sentencias para la captura y manejo de errores de operación SQL utilizando el entorno Node.js.</p>	<p>4. Transaccionalidad: Transacciones en bases de datos. Qué es una transacción de base de datos. Por qué se necesita control de transacciones. La integridad de datos. Cuándo usar transacciones. Ejemplos de uso común de transacciones. Instrucciones Begin, Commit y Rollback. Algorítmica habitual de una transacción. La propiedad ACID (Atomicidad, Consistencia, Aislamiento, Permanencia). Journaling y Bloqueos. Operaciones transaccionales. Ejecutar múltiples operaciones transaccionales con instrucción COMMIT. Ejecutar múltiples operaciones transaccionales con instrucción ROLLBACK. La instrucción AUTOCOMMIT. Captura de errores y operaciones transaccionales. Errores de operación de datos. Captura de errores en Node. Procesando errores y condiciones para la operación de transacciones en Node: Errores en tiempo de ejecución, Errores controlados, Condiciones de borde. Operaciones de desconexión y errores generados por el Motor de base de datos.</p>
5. Implementar la capa de acceso a datos utilizando un ORM con entidades no relacionadas para realizar operaciones CRUD.	<p>5.1 Reconoce las características del lenguaje Javascript ES6 para la definición de clases y objetos.</p> <p>5.2 Identifica las características, ventajas y desventajas de utilizar un ORM para la implementación de la capa de acceso a datos en un aplicativo Node.js.</p> <p>5.3 Define modelos que representan una entidad especificando propiedades, tipos de datos y otras</p>	<p>5. Acceso a datos con ORM: Clases en ES6. Qué es una Clase. Definición de una clase por medio de una declaración. Definición de una clase por medio de una expresión. El método constructor. Accesadores y mutadores. Métodos personalizados. Instanciación de una clase. Mapeo de Objeto-Relacional (ORM): Qué es un ORM, Por qué utilizarlo, Ventajas y desventajas. Qué</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

	<p>opciones acorde a la librería Sequelize.</p> <p>5.4 Implementa operaciones CRUD en un programa Node.js para la manipulación de datos acorde a la librería Sequelize.</p>	<p>es un modelo. Qué son las relaciones Sequelize ORM. Qué es Sequelize. Instalación de Sequelize. Configuración básica de Sequelize. Definición del modelo en Sequelize. Definición de atributos y tipos de datos. Realizando operaciones CRUD (creación, lectura, actualización y eliminación).</p>
6. Utilizar asociaciones uno a uno, uno a muchos y muchos a muchos en la definición de las relaciones entre las entidades de un modelo que resuelven un problema.	<p>6.1 Identifica el tipo de asociación entre entidades para representar un problema determinado.</p> <p>6.2 Utiliza sentencias para definir la asociación entre las entidades de un modelo acorde a la librería Sequelize.</p> <p>6.3 Utiliza sentencias para la lectura y despliegue de objetos relacionados en un programa Node.js de acuerdo a los requerimientos.</p>	<p>6. Manejo de relaciones en un ORM:</p> <p>Relaciones Uno a Muchos. Qué es una relación muchos a uno. Cuándo se utiliza. Cómo se implementa. Definición de borrado en cascada</p> <p>Relaciones Muchos a Muchos. Qué es una relación muchos a muchos. Cuándo se utiliza. Cómo se implementa. Definiendo entidades intermedias con más campos</p> <p>Relaciones Uno a Uno. Qué es una relación uno a uno. Cuándo se utiliza. Cómo se implementa en un modelo.</p>
PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
<p>*Formación profesional en el área de la ingeniería informática, electrónica, industrial, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web con Javascript, el entorno Node y base de datos PostgreSQL, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Formación académica como técnico de nivel superior en el área informática, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web con Javascript, el entorno Node y base de datos PostgreSQL, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Experiencia laboral de mínimo 6 años desempeñando funciones de desarrollo web con Javascript, el entorno Node y base de datos PostgreSQL, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		
Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p> <p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes.</p> <p>*Equipo computacional, uno para cada participante, al menos Pentium Core 5, 8 GB RAM y 500 GB HD, y conexión a internet, con el siguiente software instalado:</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p> <p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p> <p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.	Node 10.X o superior compatible; Visual Studio Code Editor 1.3 o superior; PostgreSQL 12.0 o superior	curso
--	---	-------

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



MÓDULO FORMATIVO N° 8		
Nombre	IMPLEMENTACIÓN DE API BACKEND NODE EXPRESS	
N° de horas asociadas al módulo	46,00	
Código Módulo	MA03007	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Licencia de Enseñanza Media completa; Manejo computacional a nivel usuario; Aprobación de test de selección en etapa I y II, consistente en prueba en línea que mide habilidades de raciocinio lógico y comprensión verbal, más un test de aptitud. **Se exceptúan de este requisito, a los postulantes al programa Talento Digital para Chile, que estén cursando cuarto año medio en liceos científico-humanista o en el caso de alumnos de liceos técnico profesional que estén cursando cuarto medio o desarrollando su práctica laboral."	
Competencia del módulo	Implementar aplicaciones empresariales que disponibilizan servicios Rest utilizando el framework Express y el entorno Node.js para dar solución a los requerimientos de la organización	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Describir las características principales de una arquitectura REST distinguiendo buenas prácticas para el diseño de una API REST para la interoperación de sistemas.	1.1 Reconoce las características principales de una arquitectura REST para la interoperación de sistemas. 1.2 Reconoce las buenas prácticas para la construcción de una API Rest.	1. APIs RESTful: Qué significa REST y para qué sirve. Características principales. Ventajas y desventajas. Reglas de una arquitectura REST: Interfaz uniforme con mensajes descriptivos, Peticiones sin estado, Cacheable, Separación cliente, Sistema de capas. Buenas prácticas en la creación de una API REST. Versionamiento de la api. Verbos HTTP. Nombre de los recursos. Códigos de estado. Formato de salida. Búsqueda y filtrado. HATEOAS.
2. Implementar un servidor REST utilizando el framework Express para la disponibilización de recursos acorde a las buenas prácticas.	2.1 Implementa rutas o endpoints para reflejar el estado representacional de un recurso utilizando las buenas prácticas de diseño acorde al framework Express. 2.2 Implementa el cuerpo de una respuesta a una petición utilizando las buenas prácticas de diseño acorde al framework express. 2.3 Utiliza códigos de respuesta a una petición acorde a las buenas prácticas de diseño.	2. Implementando una API REST: Iniciar un proyecto REST Server con Express. Crear Rutas con retorno JSON para peticiones HTTP del tipo: GET, PUT, POST, DELETE. En qué caso se usa cada tipo de Petición. Cómo es la estructura de los Endpoints según el tipo de petición. Recepción de parámetros en una petición HTTP. Procesando el retorno JSON. Uso del paquete body-parser. Códigos de respuesta HTTP. Inspeccionar los distintos códigos de respuesta del protocolo HTTP: 1XX, 2XX, 3XX, 4XX, 5XX. Respondiendo códigos personalizados en una petición HTTP.
3. Implementar la funcionalidad de subida de archivos a un servidor utilizando Express de acuerdo al entorno Node.js.	3.1 Reconoce los paquetes requeridos para la implementación de upload en un servidor Express. 3.2 Implementa funcionalidad de subida de archivos en	3. Subida de archivos al servidor: Uso del paquete express-fileupload, Instalación. Configuración. Creando una aplicación para subir archivos al servidor. Validando

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

	<p>un servidor Express.</p> <p>3.3 Utiliza archivos variados subidos por Upload File en un proyecto Express.</p>	<p>existencia del archivo. Validando extensiones permitidas. Recibiendo archivos. Obteniendo y modificando el nombre de archivo. Retornando mensajes de estado al cliente. Obteniendo atributos propios del archivo. Eliminando archivos del servidor.</p>
<p>4. Implementar mecanismos de seguridad a un servicio REST de un servidor Express utilizando JWT de acuerdo al entorno Node.js.</p>	<p>4.1 Identifica las componentes principales del protocolo JWT y la estructura básica de la mensajería para la securización de servicios REST.</p> <p>4.2 Reconoce la necesidad del uso de seguridad en servicios Web y los mecanismos necesarios para su implementación.</p> <p>4.3 Implementa un servicio para la obtención de un token de usuario utilizando Express y Node.js.</p> <p>4.4 Implementa validación de token en las peticiones de una servicio REST acorde a las buenas prácticas.</p>	<p>4. Securización mediante JWT: Qué es JWT: Por qué debemos agregar seguridad a nuestros servicios. El modelo stateless. Seguridad con JWT. Firma de token. Clave del servidor. Proceso de autenticación. Tiempo de vida y caducidad de token. JWT v/s variables de sesión y cookies?. Componentes de un JWT. Header. Algoritmos de encriptación. Payload. El elemento iat. SignatureEl código Base64: Almacenamiento en LocalStorage/SessionStorage. Revisión de algunos de los campos estándares de la mensajería Implementación en Express. Instalando el paquete jsonwebtoken. Instalando el paquete express-jwt. Configurar el paquete. Crear un proyecto básico con seguridad JWT para generar un token. Validar el token en llamadas HTTP. Invocando y servicio pasando un token como parámetro. Decodificando Tokens . Validar token.</p>
PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
<p>*Formación profesional en el área de la ingeniería informática, electrónica, industrial, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web con Javascript, el entorno Node y Express, Desarrollo de API Rest con Node y Express, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Formación académica como técnico de nivel superior en el área informática, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de desarrollo web con Javascript, el entorno Node y Express, Desarrollo de API Rest con Node y Express, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Experiencia laboral de mínimo 6 años desempeñando funciones de desarrollo web con Javascript, el entorno Node y Express, Desarrollo de API Rest con Node y Express, demostrable.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		
Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla.</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

<p>Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.</p>	<p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes.</p> <p>* Equipo computacional, uno para cada participante, al menos Pentium Core 5, 8 GB RAM y 500 GB HD, y conexión a internet, con el siguiente software instalado: Node 10.X o superior compatible; Visual Studio Code Editor 1.3 o superior; Google Chrome 70.0+ o Mozilla Firefox 64.0+; Postman 7.9 o superior</p>	<p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p> <p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del curso</p>
---	--	---

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



MÓDULO FORMATIVO N° 9		
Nombre	DESARROLLO DE PORTAFOLIO DE UN PRODUCTO DIGITAL	
N° de horas asociadas al módulo	18,00	
Código Módulo	MB00132	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Según el consignado en el plan formativo.	
Competencia del módulo	Presentar un producto digital en un portafolio personal utilizando las herramientas tecnológicas y buenas prácticas disciplinares para resolver una problemática y potenciar el perfil profesional	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Reconocer las características fundamentales de un portafolio de productos digitales efectivo para potenciar el perfil profesional.	1.1 Reconoce la importancia de un portafolio de productos como herramienta para potenciar el perfil profesional. 1.2 Describe las buenas prácticas para la creación de un portafolio de productos. 1.3 Identifica herramientas para la implementación de un portafolio de productos.	1. Portafolio de Productos: El portafolio de productos. Qué es un portafolio de productos. Importancia de contar con un portafolio. Buenas prácticas para la creación de un portafolio de productos. Herramientas que se pueden utilizar para la creación del portafolio de productos: GitHub, Hosting, Página web personal, Youtube, Otros.
2. Elaborar un producto tecnológico para resolver una problemática real utilizando las buenas prácticas de la disciplina.	2.1 Distingue conceptos y buenas prácticas para el diseño de un producto digital que resuelve un problema. 2.2 Utiliza técnicas y herramientas de la disciplina para la implementación de un producto digital. 2.3 Implementa un producto funcional que resuelve un problema real utilizando las buenas prácticas de la disciplina.	2. Finalización del proyecto: Revisión del producto construido a lo largo del curso. Depuración y mejora del producto. Feedback y retroalimentación. Ajustes finales y cierre de entregable.
3. Implementar un portafolio de un producto digital utilizando las herramientas disponibles y las buenas prácticas de la industria.	3.1 Utiliza GitHub/Behance para la creación de un portafolio de productos digitales utilizando las buenas prácticas de la industria. 3.2 Utiliza un servicio de hosting para alojar un producto tecnológico de acuerdo a su naturaleza. 3.3 Presenta un producto digital utilizando la plataforma youtube para demostrar las capacidades técnicas.	3. Herramientas para la implementación de un portafolio: Utilización de GitHub para crear un portafolio (*). Qué es GitHub. Características y operaciones básicas de GitHub. Buenas prácticas para tener un portafolio atractivo en GitHub. Buenas prácticas para la página principal de un repositorio. Buenas prácticas para la página de perfil. Utilización de Behance para crear un portafolio (**). Qué es Behance. Características y operaciones básicas de Behance. Buenas prácticas para tener un portafolio atractivo en Behance. Behance dedicado y exclusivo a proyectos de UX. Alojamiento de tu producto en un servidor. Qué es un Hosting. Servicios gratuitos de hosting. Servicios cloud gratuitos. Cómo

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



		alojar un proyecto UX/UI. Cómo alojar un proyecto Front-End. Cómo alojar un proyecto Fullstack Java. Cómo alojar un proyecto Fullstack Javascript. Cómo alojar un proyecto Fullstack Python. Cómo alojar un proyecto Android. Utilización de Youtube. Utilización de youtube en el mundo del desarrollo de productos digitales. Herramientas gratuitas para la confección de un video técnico. Cómo confeccionar un tutorial técnico en youtube para demostrar tus capacidades. Cómo presentar tu proyecto en youtube. (*) Aplica sólo para cursos de desarrollo. (**) Aplica sólo para curso de diseño UX/UI.
PERFIL DEL FACILITADOR		
Opción 1	Opción 2	Opción 3
<p>*Formación profesional en el área de las ingenierías para el caso de los planes formativos front-end, fullstack, mobile, o bien como diseñador para el caso del plan UX/UI. En todos los casos anteriores, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de laborales ya sea como desarrollador front-end, fullstack, mobile, o diseñador UX/UI, dependiendo del plan formativo, o bien gerenciando áreas en estas materias.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Formación académica como técnico de nivel superior en el área informática para el caso de los planes formativos front-end, fullstack, mobile, o bien como diseñador para el caso del plan UX/UI. En todos los casos anteriores, con título.</p> <p>*Experiencia laboral de mínimo 3 años desempeñando funciones de laborales ya sea como desarrollador front-end, fullstack, mobile, o diseñador UX/UI, dependiendo del plan formativo, o bien gerenciando áreas en estas materias.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>	<p>*Experiencia laboral de mínimo 6 años desempeñando funciones de laborales ya sea como desarrollador front-end, fullstack, mobile, o diseñador UX/UI, dependiendo del plan formativo, o bien gerenciando áreas en estas materias.</p> <p>*Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.</p>
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		
Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p> <p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes.</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p> <p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p> <p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.		curso
--	--	-------

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



MÓDULO FORMATIVO N° 10		
Nombre	DESARROLLO DE EMPLEABILIDAD EN LA INDUSTRIA DIGITAL	
N° de horas asociadas al módulo	18,00	
Código Módulo	MB00133	
Perfil ChileValora asociado al módulo	SIN PERFIL CHILEVALORA ASOCIADO.	
UCL(s) ChileValora relacionada(s)	SIN UCL ASOCIADA.	
Requisitos de ingreso	Según el consignado en el plan formativo.	
Competencia del módulo	Elaborar un plan de búsqueda desarrollando el propio perfil profesional, reconociendo las características del mercado laboral del sector tecnológico y aplicando técnicas para la preparación de entrevistas	
Adaptabilidad a modalidad no presencial	ESTE MÓDULO PUEDE SER ADAPTADO POR EL EJECUTOR PARA DESARROLLO ONLINE.	
APRENDIZAJES ESPERADOS	CRITERIOS DE EVALUACIÓN	CONTENIDOS
1. Explicar los elementos fundamentales de un plan de búsqueda laboral distinguiendo las características de la industria TI en el proceso de selección.	1.1 Reconoce las características del mercado laboral TI para la búsqueda de empleo. 1.2 Describe los elementos que constituyen un plan de búsqueda efectivo para el mercado laboral TI. 1.3 Describe las características principales de un perfil profesional para la industria TI distinguiendo competencias técnicas y habilidades personales valoradas. 1.4 Reconoce en qué consiste un desafío técnico como medio para potenciar el perfil profesional. 1.5 Reconoce los principales elementos a considerar para preparar una entrevista laboral.	1. El plan de búsqueda laboral: Características del mercado laboral TI. Quiénes buscan perfiles como el tuyo. Cómo se realiza la búsqueda de candidatos. Etapas del proceso de selección. Qué tipo de empresas y áreas contratan a estos perfiles, qué tipo de cargos están disponibles (cómo se llaman). Cómo es la estructura organizacional en trabajos TI, en empresas TI y resto, por ejemplo con quién trabaja, cómo trabaja, tipos de contratos más frecuentes (incluir freelance y externalización en la industria). El plan de búsqueda laboral. Qué es un plan de búsqueda laboral. Por qué debemos planificar la búsqueda. Elementos de un plan de búsqueda: El perfil profesional. El currículum vitae. Publicación y difusión del perfil. El portafolio de productos. Los desafíos técnicos: Las entrevistas laborales, Los Test Psicolaborales. El perfil profesional. Qué es un perfil profesional. Aspectos claves para la confección de un perfil profesional. Competencias técnicas valoradas por la industria TI vs las personales (cuáles son mis fortalezas y debilidades). Habilidades personales valoradas por la industria TI vs las personales (cuáles son mis fortalezas y debilidades). Niveles de experiencia y seniority del perfil. Principales líneas de desarrollo de carrera vs perfil propio. Los desafíos técnicos: Qué es un desafío técnico, Qué debes demostrar en un desafío técnico. Las entrevistas laborales: En qué consiste una entrevista laboral, Cómo preparar una entrevista individual. Los test

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



		psicolaborales: Qué es un test psicolaboral y para qué sirve, Recomendaciones para enfrentar un test psicolaboral.
2. Crear un currículum vitae para la contratación en la industria TI utilizando los diversos canales de publicación y empleando las buenas prácticas y recomendaciones de la industria.	2.1 Describe los elementos relevantes de un currículum vitae para la búsqueda laboral en la industria TI. 2.2 Reconoce los mecanismos para la publicación del perfil profesional en portales laborales para el área TI. 2.3 Crea un currículum vitae para la búsqueda laboral en el mercado TI utilizando buenas prácticas.	2. Confección y publicación del currículum vitae: Confección del Currículum Vitae. Qué es un Currículum Vitae. Elementos que lo componen. Reglas de oro para la confección de un CV efectivo. Buenas prácticas para la redacción de la experiencia. Buenas prácticas para la redacción de los datos educacionales. Selección y redacción de tus habilidades. Secciones adicionales en el currículum. Publicación de tu perfil en portales laborales. Sacarle el máximo provecho a LinkedIn. Conociendo Trabajando. Conociendo Laborum. Conociendo GetOnBoard. Conociendo FirstJob. Empresas headhunters. Autoempleo y freelance.
3. Aplicar técnicas para sostener entrevistas laborales efectivas reconociendo los elementos relevantes de contexto de la industria TI.	3.1 Describe el rol de las emociones en la búsqueda laboral distinguiendo mecanismos para gestionarlos. 3.2 Reconoce los elementos fundamentales de la escucha activa para el éxito en los procesos de selección. 3.3 Describe los elementos críticos para la preparación de una entrevista laboral. 3.4 Elabora un elevator pitch del perfil profesional para enfrentar una entrevista laboral. 3.5 Aplica técnicas y buenas prácticas para sostener una entrevista laboral.	3. Cómo enfrentar entrevistas laborales: Gestión emocional al servicio de la búsqueda de oportunidades laborales. Qué son las emociones. Emociones y estados de ánimo básicos. La inteligencia emocional. El rol que juegan las emociones en la búsqueda laboral. Cómo gestionar las emociones. Escucha Activa. El modelo de escucha activa. Acotar la brecha comunicacional. El contacto visual. La comunicación no verbal. La técnica de storytelling. Las entrevistas laborales. En qué consiste una entrevista laboral. Tipos de entrevista: Individual, grupal, Dirigida, libre, mixta. Cómo preparar una entrevista individual: Realizar una investigación previa, La vestimenta, El saludo, La puntualidad y el descanso, Cómo referirte al entrevistador, La postura corporal, El lenguaje y la forma de hablar, La despedida. Preparación de un Elevator Pitch. Qué es un Elevator Pitch. Cómo confeccionar un elevator pitch para una entrevista laboral.
PERFIL DEL FACILITADOR		
Unica Opción		
*Formación profesional en el área de la psicología laboral, administración, recursos humanos o ingenierías, con título. *Experiencia laboral de al menos 5 años en el área de recursos humanos y/o selección y reclutamiento. *Experiencia como facilitador/a de capacitación para personas adultas, y/o bien como docente en instituciones de educación superior, sean universidades, institutos profesionales o centros de formación técnica, de mínimo 100 horas cronológicas, demostrable.		
RECURSOS MATERIALES PARA LA IMPLEMENTACIÓN DEL MÓDULO FORMATIVO		

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE



Infraestructura	Equipos y herramientas	Materiales e insumos
<p>*Laboratorio computacional que cuente al menos con 1,5 m2 por participante, implementada con: Puestos de trabajo individuales que considere mesa y silla. Escritorio y silla para el facilitador.</p> <p>*Conexiones para utilizar medios didácticos tales como data y salida a internet.</p> <p>*Sistema de iluminación y ventilación adecuados</p> <p>*Servicios higiénicos separados para hombres y mujeres en recintos de aulas y de actividades prácticas.</p> <p>*Para actividades E-Learning debe contar con plataforma LMS implementada y acceso a sistemas de video llamada para sesiones sincrónicas, que aseguren el contacto y la conectividad de participantes.</p>	<p>*Notebook o PC para facilitador.</p> <p>*Proyector multimedia.</p> <p>*Telón.</p> <p>*Pizarrón.</p> <p>*Filmadora o cámara fotográfica para registrar evidencias de actividades realizadas por los participantes.</p>	<p>*Set de artículos de escritorio por participante (lápiz pasta, lápiz grafito, goma de borrar, regla, cuaderno o croquera, archivador o carpeta).</p> <p>*Material de apoyo y guía sobre el sector.</p> <p>*Plumones para pizarrón.</p> <p>*Libro de clases.</p> <p>*Pautas de evaluación por actividad.</p> <p>*Manual del participante que contemple todos los contenidos especificados para este módulo.</p> <p>*Material audiovisual que contenga información del curso</p>

EL CONTENIDO DE ESTE DOCUMENTO NO PUEDE SER MODIFICADO SIN LA AUTORIZACIÓN DE SENCE

