

RESOLUCIÓN (CS) N°: 16 ABR 2019

025 - 19

FLORENCIO VARELA,

Visto las Leyes Nros. 24.521 y 26.576, la Resolución ME N° 1154/10, el Acta CS N° 001/13, el Expediente N° 3267/18 del Registro de la UNIVERSIDAD NACIONAL ARTURO JAURETCHE, y

CONSIDERANDO:

Que mediante el Expediente citado en el Visto se tramita el dictado del Curso de Formación Profesional de Programador que se dictarán en forma conjunta entre el Centro de Política y Territorio, el Instituto de Ingeniería y Agronomía de la UNIVERSIDAD NACIONAL ARTURO JAURETCHE y la MUNICIPALIDAD DE FLORENCIO VARELA.

Que mediante Resolución (CS) N°086 de fecha 4 de septiembre de 2017 se creó el Centro de Formación en Oficios y Certificación de Competencias dependiente del Centro de Política y Territorio.

Que la Universidad tiene como misión primaria contribuir al desarrollo económico, social y cultural de la región, a fin de mejorar su calidad de vida y fortalecer los valores democráticos en el conjunto de la sociedad, articulando el conocimiento universal con los saberes producidos por nuestra comunidad.

Que el desarrollo de la vinculación es una de las funciones básicas de la Universidad, junto con la docencia y la investigación.

Que la UNIVERSIDAD NACIONAL ARTURO JAURETCHE se ha propuesto impulsar las actividades de vinculación como una manera de contribuir al desarrollo del territorio de su influencia, a través de la articulación y cooperación entre los distintos

8

productores del saber, transformando la información en conocimiento y atendiendo a las demandas sociales.

Que el Curso de Formación Profesional de Programador será implementado en el marco de las acciones acordadas con la Municipalidad de Florencio Varela mediante el Convenio Específico tramitado por el Expediente N° 3240/2018 del registro de la UNAJ.

Que para llevar a cabo el mencionado curso de vinculación es necesaria la designación de un coordinador.

Que el mismo ha sido aprobado por el CONSEJO SUPERIOR de esta Universidad en su sesión de fecha 16 de abril del 2019, según consta en Acta CS N° 2/19.

Que la presente medida se dicta en ejercicios de las facultades y competencias conferidas en el "Estatuto de la UNIVERSIDAD NACIONAL ARTURO JAURETCHE".

Por ello,

EL CONSEJO SUPERIOR DE LA UNIVERSIDAD NACIONAL ARTURO JAURETCHE

RESUELVE:

ARTÍCULO 1º: Aprobar el Curso de Formación Profesional de Programador que se dictará en forma conjunta entre el Centro de Política y Territorio, el Instituto de Ingeniería y Agronomía de la UNIVERSIDAD NACIONAL ARTURO JAURETCHE y la Municipalidad de Florencio Varela, cuyo programa se adjunta como Anexo Único de la presente.


ARTÍCULO 2º: Aprobar la designación del Dr. Ing. Martín Morales como Coordinador Académico del Curso de Formación Profesional.

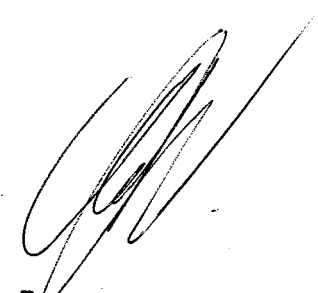


ARTICULO 3º: Autorizar a la SECRETARÍA ECONÓMICO-FINANCIERA de la UNIVERSIDAD NACIONAL ARTURO JAURETCHE a disponer de los fondos relacionados para la ejecución del Curso de Formación Profesional.

ARTICULO 4º: Regístrese, comuníquese y archívese.

RESOLUCION (CS) N°: **025-19**

  
Lic. Prof. Rafael Ruffo  
DIRECTOR  
Centro de Política y Territorio  
Universidad Nacional Arturo Jauretche

  
Lic. Ernesto F. Villanueva  
RECTOR  
Univ. Nac. Arturo Jauretche

ANEXO ÚNICO RESOLUCIÓN (CS) N°: **025-19**

## CENTRO DE FORMACIÓN EN OFICIOS Y ACREDITACIÓN DE COMPETENCIAS

### FICHA RESUMEN DE LA PROPUESTA FORMATIVA:

- Curso: Curso Programador
- Sector/es de actividad socio productiva: Informática - Software y Servicios Informáticos
- Denominación del perfil profesional: PROGRAMADOR
- Familia profesional: INFORMÁTICA
- Ámbito de la trayectoria formativa: FORMACIÓN PROFESIONAL
- Tipo de certificación: CERTIFICADO DE FORMACIÓN PROFESIONAL INICIAL
- Nivel de la Certificación: III
- Carga horaria: 388 horas.
- Referencia de ingreso: Acreditar nivel secundario aprobado
- Marco de referencia: Res CFE N° 289/16 Anexo I

### FUNDAMENTACIÓN:

El sector profesional del desarrollo del software es un sector en auge con un gran crecimiento sostenido en estos últimos años debido a la gran demanda en el desarrollo de software de calidad adaptable a las últimas tecnologías.

En la región, la demanda requerida para el desarrollo es mayor a la de oferta disponible, por lo que es de vital importancia la formación de nuevos profesionales que puedan afrontar a las necesidades de la población de la región en relación a la informática.

El Centro de Formación en Oficios y Acreditación de Competencias Laborales de la UNAJ se propone ofrecer capacitación de calidad en oficios para el fortalecimiento de lazos con el

mundo laboral o inserción en el mismo a población fundamentalmente de los distritos de Florencio Varela, Quilmes, Berazategui y Almirante Brown que vea vulnerado sus derechos al trabajo y a la educación (jóvenes desocupados, sub-ocupados, trabajadores de la economía informal, personas con niveles de escolaridad obligatorios incompletos).

Para personas sin empleo o en situación de precariedad laboral, el acceso a propuestas formativas de calidad se presenta como una oportunidad y una alternativa de mejora en sus trayectorias laborales.

En el curso se abordarán los conocimientos generales de distintos aspectos de la Informática con el fin de poder desarrollar un conjunto de competencias específicas de desarrollo de software dentro del sector profesional de Informática.

## PERFIL PROFESIONAL<sup>1</sup>

El perfil profesional del Programador alude al conjunto de actividades en las que el profesional puede demostrar, una vez que ha logrado su certificación, los conocimientos adquiridos sobre su profesión. Este perfil involucra un conjunto de competencias específicas de desarrollo de software dentro del sector profesional de Informática. Se articula en torno a núcleos curriculares a partir de las demandas socio-productivas del sector y la realidad del medio industrial y de servicios.

### Alcance del perfil profesional

El Programador está capacitado, de acuerdo a las actividades que se detallan en el Perfil Profesional, para escribir, testear, depurar y mantener el código de programación de los programas en el marco de un equipo de trabajo o en forma individual bajo supervisión. Los programadores escriben los programas de acuerdo a las especificaciones formales dadas por los analistas de sistemas. Ellos transforman el diseño en instrucciones concretas que pueden ejecutarse en la computadora. También tienen como tarea testear el programa en busca

  
<sup>1</sup> Toma como referencia lo establecido en la RE del CFE N° 289/16 Anexo I correspondiente a la denominación Programador

de bugs (errores de programación) dentro del proceso de prueba y depuración de errores. El software necesita ser actualizado, reparado, modificado y ampliado, es decir mantenido, siendo esta tarea realizada en parte por los programadores. Este profesional requiere supervisión en todas las actividades que desarrolla. Siempre reporta a superiores y se remite a ellos para solicitar instrucciones sobre su desempeño.

Funciones que ejerce el profesional

1. Escribir código de programación de acuerdo a especificaciones formales.

El Programador interpreta las especificaciones de diseño o requisitos de las asignaciones a programar comprendiendo en su contexto inmediato cuál es el problema a resolver, determinando el alcance del problema y convalidando su interpretación a fin de identificar aspectos faltantes. Desarrolla algoritmos que dan soluciones a los problemas asignados o derivados de los mismos y efectúa pruebas de unidad de los módulos para asegurar que cumplan con las especificaciones del diseño a fin de tener un código eficiente, documentado, fácil de entender y mantener.


2. Interpretar especificaciones de diseño de las asignaciones a programar en el contexto del desarrollo de software en el que participa.

El Programador recibe las especificaciones formales o informales del Líder de Proyecto y analiza el problema a resolver; interpreta el material recibido y clarifica eventuales malas interpretaciones o desacuerdos convalidando su interpretación con el responsable del proyecto.

3. Planificar su trabajo en el contexto del equipo de desarrollo del proyecto.

El Programador establece secuencia y orden de trabajo, identifica prioridades y tiempos de consulta para aspectos no previstos, aplicando metodologías de resolución de problemas para la planificación y organización de su trabajo.

4. Verificar el código desarrollado.

 El Programador determina las necesidades de cobertura en función de las características y normas de calidad del proyecto, identifica las clases de equivalencia de datos utilizados in-

ternamente o intercambiados y ejecuta los casos de prueba. Realiza las pruebas correspondientes, registrando los datos y resultados alcanzados, así como las acciones correctivas realizadas para solucionar las fallas encontradas.

5. Depurar estructuras lógicas o códigos de programas.

El Programador relaciona resultados incorrectos con los datos o porciones de código que los originaron, analiza estos datos y/o partes del código que causaron el mal funcionamiento y determina el tipo de corrección o reemplazo requeridos; verifica que la corrección y/o reemplazo solucionen el mal funcionamiento o la salida de resultados erróneos.

6. Manejo y manipulación de los datos y su relación con las aplicaciones desarrolladas o a desarrollar.


El Programador conoce la estructura de los datos, su organización, la relación entre entidades y su uso en las aplicaciones desarrolladas o por desarrollar. También consulta a pares y al líder del equipo de trabajo para reflexionar y recibir ayuda que le permita resolver problemas relacionados con el manejo de los datos o aporta sus conocimientos a otros.

7. Realizar la documentación técnica y de usuarios de acuerdo con los requerimientos funcionales y técnicos recibidos.

El Programador realiza la documentación describiendo qué hace cada parte del código y por qué se incluyen, qué datos o situación lo originaron; registros y evidencias de las actividades realizadas y de los incidentes observados, debiendo identificar cada versión de acuerdo a estándares establecidos dentro del equipo de desarrollo.

### Área Ocupacional

Este profesional se inserta ocupacionalmente en distintos tipos de organizaciones donde se realiza desarrollo de software, ya sea por requerimiento de otras organizaciones locales o extranjeras o por solicitud de usuarios particulares.

 Entre dichas organizaciones se cuentan empresas que proveen software conjuntamente con servicios de asesoramiento y consultoría, y que desarrollan sus propios productos de softwa-

re para vender en el país o en el exterior. También en organizaciones dedicadas a otras actividades, pero que producen el software que necesitan para desarrollar sus propias actividades o que integran en productos que venden.

El software debe satisfacer las especificaciones de los requerimientos, ya sean formales o informales, las que pueden venir dadas por el cliente, algún consultor especializado en el tipo de problemas que aborda la aplicación o ser elaboradas por algún analista funcional integrante del equipo de trabajo del proyecto.

El equipo de desarrollo suele estar encabezado por un gerente o líder, quien es responsable por el proyecto. Integra diversos roles ocupacionales, como el de arquitecto de software, que establece el diseño general del sistema y especificaciones de calidad de la solución, así como una serie de programadores que son quienes realizan la codificación de acuerdo a las especificaciones recibidas. Del equipo de trabajo pueden participar uno o más analistas técnicos que se ocupan de detalles relativos a aspectos de tecnología, seguridad, bases de datos o estándares de programación y asesoran y dan apoyo técnico a los programadores. Eventualmente pueden participar diseñadores gráficos y especialistas en otros aspectos específicos.

La posición ocupacional de la figura que es referencia del presente trayecto es la que suele denominarse Programador. Integra equipos de proyecto dedicados al desarrollo o mantenimiento de software y recibe asignaciones específicas que debe resolver en lapsos que suelen medirse en términos de días o semanas, produciendo módulos o programas que satisfagan especificaciones y se integren al sistema objeto del proyecto.

A partir de especificaciones de diseño y del conocimiento de la arquitectura del sistema, los programadores construyen la parte que les fuera asignada en base a desarrollos ya existentes y adaptando o escribiendo uno nuevo para cumplir con los requerimientos. Testean su trabajo, verifican unitariamente lo producido y lo entregan para ser probado integralmente e integrado al resto.

Habitualmente pueden estar especializados en una tecnología determinada, trabajan individualmente o de a pares dentro de un grupo más numeroso, brindándose mutuamente cola-



boración para resolver los problemas que deben enfrentar. Resuelve estas asignaciones individualmente o trabajando en pares, recibiendo la supervisión y asesoramiento de un líder de proyecto con quien consulta dudas y decisiones significativas o comunica inconvenientes. También recibe apoyo y brinda colaboración a otros miembros del grupo.

También puede desempeñarse en forma autónoma, asumiendo la mayor parte de las tareas propias del proceso, sobre todo trabajando en forma independiente resolviendo problemas de pequeñas organizaciones que requieren sistemas de baja complejidad y reducida dimensión.

De lo anterior se desprende que el Programador desarrolla su actividad en las siguientes áreas ocupacionales:

- Servicios informáticos para pequeñas y medianas empresas en áreas de desarrollo y producción de software.
- Empresas de distintos sectores de actividad económica en áreas de informática o de procesamiento de datos.
- Por cuenta propia o en pequeños emprendimientos asociativos de desarrollo y producción de software.
- Empresas de servicios de mantenimiento de sistemas informáticos.
- Comercialización de equipos y sistemas informáticos.
- Administración pública, en las áreas de mantenimiento y gestión de la información.
- ONGs, en áreas vinculadas con el procesamiento de datos para la gestión.
- Mantenimiento de sistemas informáticos en entornos personales.
- Asesoramiento técnico y venta de sistemas y aplicaciones informáticas.

#### OBJETIVOS:

##### Objetivo General:

Que los estudiantes adquieran las competencias y capacidades necesarias para desempeñarse en el desarrollo profesional del software, en cualquiera de los ámbitos alcanzados por el curso así como continuar capacitándose para lograr un crecimiento y desarrollo profesional.

Objetivos Específicos:

Se espera que los estudiantes al finalizar el curso puedan:

- Conocer los elementos básicos de la informática, cuales son las principales áreas y en qué consisten.
- Interpretar diagramas y desarrollar código en base a los mismos.
- Desarrollar aplicaciones utilizando el paradigma POO en base a los requerimientos dados.
- Trabajar en equipos profesionales interdisciplinarios, con la capacidad de transmitir y comunicar lo realizado.
- Realizar desarrollos propios, con la posibilidad de emprender una idea de software y llevarla a cabo.
- Realizar buenas prácticas de programación para obtener código de calidad.
- Documentar el código hecho.
- Interpretar y confeccionar diagramas DER para el diseño de bases de datos.
- Manipular, administrar y utilizar bases de datos relacionales.

Además se espera fomentar el desarrollo del software en la región.

CAPACIDADES PROFESIONALES-COMPETENCIAS Y CONTENIDOS POR MÓDULO:

Capacidades profesionales/competencias:

- Interpretar información, escrita o verbal, relacionada con especificaciones de diseño, identificando simbología propia de la actividad, verificando su pertinencia y alcance para realizar una acción requerida.
- Transferir la información de los documentos a su tarea, relacionada con desarrollo de software, verificando su pertinencia y alcance para realizar una acción requerida a partir de la especificación de diseño recibida.



- Establecer relaciones de cooperación, coordinación e intercambio en el propio equipo de trabajo y con otros equipos de desarrollo de software que intervengan con el equipo de trabajo.
- Transmitir información técnica de manera verbal, sobre el desarrollo de piezas de software, que le fueron encomendadas.
- Gestionar y administrar sus propios recursos (principalmente tiempos), necesarios para el avance de su propio trabajo y el trabajo en equipo, según las condiciones establecidas por los responsables de las tareas encomendadas.
- Gestionar la relación comercial que posibilite la obtención de empleo y las relaciones que devengan con los prestadores de servicios.
- Identificar los problemas que se presenten en el desarrollo o programación a partir del análisis, jerarquización y priorización de la información.
- Integrar las técnicas de trabajo, la información, la producción de códigos y reutilización de códigos existentes, el armado de estructuras de datos, la construcción y modificación de clases y objetos, el diseño de interfaces de usuario, los criterios de calidad de software en las acciones de desarrollo propias de su actividad.
- Aplicar criterios de calidad de software (cohesión y acoplamiento) en los procesos de desarrollo de software, tendiendo a generar propuestas de mejoramiento continuo en métodos de programación, técnicas de prueba unitaria, verificación y depuración; y organización del trabajo en equipo.
- Analizar datos y su relación con la aplicación a desarrollar pudiendo determinar cuáles son necesarios a utilizarse según el momento dado.
- Administrar una base de datos utilizando una DBMS, configurando permisos, creando bases de datos, configurando copias de seguridad, utilizando DDL.
- Interpretar consultas por medio del estándar SQL, realizando distintos tipos de consultas a la base de datos.

Contenidos de la enseñanza relacionados con las capacidades profesionales-competencias:





025-19

MÓDULO	CAPACIDADES PROFESIONALES / COMPETENCIAS GENERALES	Bloques de contenidos	Contenidos
Técnicas de Programación	<p>Interpretar información, escrita o verbal, relacionada con especificaciones de diseño, identificando simbología propia de la actividad, verificando su pertinencia y alcance para realizar una acción requerida.</p> <p>Transferir la información de los documentos a su tarea, relacionada con desarrollo de software, verificando su pertinencia y alcance para realizar una acción requerida a partir de la especificación.</p>	Elementos informáticos	<ul style="list-style-type: none"> <li>● Aplicaciones y usos de la informática.</li> <li>● Hardware y Software.</li> <li>● Tecnologías informáticas libres y propietarias.</li> </ul>
		Diseño de algoritmos	<ul style="list-style-type: none"> <li>● Concepto de Programación.</li> <li>● Concepto de algoritmo.</li> <li>● Lenguajes de programación.</li> <li>● Diagramas de Flujo</li> <li>● Pruebas de Escritorio</li> <li>● Pseudocódigo</li> <li>● Definición y análisis de problemas del campo informático.</li> <li>○ Datos de entrada y salida, relación entre ellos.</li> <li>○ Variables y constantes.</li> <li>● Operadores relacionales y lógicos.</li> <li>● Operadores aritméticos.</li> </ul>

*[Handwritten signatures]*



025-19

	<p>cación de diseño recibida.</p>	<p>Desarrollo de programas</p>	<p><input type="radio"/> Desarrollo de algoritmos: Técnicas de diseño de algoritmos.</p> <p><input type="radio"/> Estructurada..</p> <ul style="list-style-type: none"> <li>● Herramientas para diseño de algoritmos.</li> </ul> <p><input type="radio"/> Diagramas.</p> <ul style="list-style-type: none"> <li>● Estructuras de programación.</li> </ul> <p><input type="radio"/> Metodología estructurada.</p> <ul style="list-style-type: none"> <li>● Estructura secuencial:             <ul style="list-style-type: none"> <li>○ Características.</li> <li>○ Definición de variables.</li> </ul> </li> <li>● Estructura alternativa             <ul style="list-style-type: none"> <li>○ Características.</li> <li>○ Expresiones lógicas.</li> </ul> </li> <li>● Estructura repetitiva o iterativa             <ul style="list-style-type: none"> <li>○ Características.</li> <li>○ Fases de un programa iterativo.</li> </ul> </li> <li>● Estructura de datos.             <ul style="list-style-type: none"> <li>○ Arrays.                     <ul style="list-style-type: none"> <li>▪ Vectores.</li> <li>▪ Matrices.</li> </ul> </li> <li>○ Listas Enlazadas.</li> <li>○ Pilas.</li> <li>○ Colas.</li> <li>○ Árboles.</li> <li>○ Grafos.</li> </ul> </li> <li>● Algoritmos de Búsqueda.</li> <li>● Algoritmos de Ordenamiento.</li> <li>● Algoritmos de Recorrido.</li> <li>● Recursividad.</li> </ul>
<p>Ingeniería de Software, Relaciones Laborales y Orientación Profesional</p>	<p>Establecer relaciones de cooperación, coordinación e intercambio en el propio equipo de trabajo y con otros equipos de desarrollo de software que intervengan con el equipo de trabajo.</p> <p>Transmitir información técnica de manera verbal,</p>	<p>Ingeniería de Software.</p>	<ul style="list-style-type: none"> <li>● Ingeniería de Software. Procesos de desarrollo.</li> <li>● Fases de desarrollo.</li> <li><input type="radio"/> Análisis y diseño</li> <li><input type="radio"/> Desarrollo</li> <li><input type="radio"/> Testing</li> <li><input type="radio"/> Mantenimiento</li> <li>● Roles. Actores en un proceso de desarrollo de Software.</li> </ul>

*R*  
*RR*



025-19

	<p>sobre el desarrollo de piezas de software, que le fueron encomendadas.</p> <p>Gestionar y administrar sus propios recursos (principalmente tiempos), necesarios para el avance de su propio trabajo y el trabajo en equipo, según las condiciones establecidas por los responsables de las tareas encomendadas.</p> <p>Gestionar la relación comercial que posibilite la obtención de empleo y las relaciones que devengan con los prestadores de servicios.</p>	<p>Derecho del Trabajo y Relaciones Laborales.</p> <p>Orientación Profesional y Formativa.</p>	<ul style="list-style-type: none"> <li>● Requerimientos</li> <li>○ Funcionales</li> <li>● No Funcionales</li> <li>● Metodologías tradicionales y enfoque ágil.</li> <li>● Repositorios.</li> <li>● Representación y negociación colectiva: Sindicatos: características organizativas. Representación y organización sindical. El convenio colectivo como ámbito de las relaciones laborales. Concepto de paritarias. El papel de la formación profesional inicial y continua en las relaciones laborales.</li> <li>● Contrato de trabajo: Relación salarial, Trabajo registrado y no registrado. Modos de contratación. Flexibilización laboral y precarización. Seguridad social. Riesgos del trabajo y las ocupaciones. La formación profesional inicial y continua como derecho de los trabajadores. La formación profesional como dimensión de la negociación colectiva y las relaciones laborales.</li> <li>● Sectores y subsectores de actividad principales que componen el sector profesional. Empresas: Tipos y características. Rasgo central de las relaciones de empleo en el sector: Ocupaciones y puestos de trabajo en el sector profesional: características cuantitativa y cualitativas. Mapa ocupacional. Trayectorias típicas y relaciones</li> </ul>
--	---	--	--

*R*  
*JK*

			funcionales. Mapa formativo de la FP inicial y continua en el sector profesional y su correspondencia con los roles ocupacionales de referencia. Regulaciones sobre el ejercicio profesional: habilitación profesional e incumbencia.
Programación Orientada a Objetos	<p>Identificar los problemas que se presenten en el desarrollo o programación a partir del análisis, jerarquización y priorización de la información.</p> <p>Integrar las técnicas de trabajo, la información, la producción de códigos y reutilización de códigos existentes, el armado de estructuras de datos, la construcción y modificación de clases y objetos, el diseño de interfaces de usuario, los criterios de calidad de software en las acciones de desarrollo propias de su actividad.</p> <p>Aplicar criterios de calidad de software (cohesión y acoplamiento) en los procesos de desarrollo de software, tendiendo a generar propuestas de mejoramiento continuo en métodos de programación, técnicas de prueba unitaria, verificación y depuración; y organización del trabajo en equipo.</p>	<p>Fundamentos de la Programación Orientada a Objetos.</p>	<ul style="list-style-type: none"> <li>● El modelo computacional de objetos.</li> <li>● Conceptos de: <ul style="list-style-type: none"> <li>○ Clase y objeto.</li> <li>○ Atributos y métodos.</li> <li>○ Estado y comportamiento.</li> </ul> </li> <li>● Mensaje entre objetos.</li> <li>● Encapsulamiento de la información.</li> <li>● Niveles de acceso.</li> <li>● Tiempo de vida de los objetos.</li> <li>● Abstracción y modularización.</li> <li>● Herencia.</li> <li>● Especialización vs generalización.</li> <li>● Superclase y subclase.</li> <li>● Clases abstractas.</li> <li>○ Polimorfismo y sobrecarga de operadores.</li> <li>● Interfaces</li> </ul>
		Metodología de Desarrollo.	<ul style="list-style-type: none"> <li>● Diseño UML.</li> <li>● Diagrama de Clases.</li> <li>○ Relaciones entre clases: herencia, asociación, composición y agregación.</li> <li>○ Diagrama de Casos de Uso.</li> <li>○ Diagrama de Secuencia.</li> <li>● Construcción de un diseño modular.</li> <li>● Acoplamiento - Cohesión.</li> </ul>
		Lenguaje de Programación Orientado a Objetos.	<ul style="list-style-type: none"> <li>● Sintaxis y estructura del lenguaje</li> <li>● Estructura de una clase</li> <li>○ Variables de instancia</li> <li>○ Métodos de acceso y de modificación</li> </ul>

			<ul style="list-style-type: none"> <li><input type="radio"/> Constructores</li> <li><input type="radio"/> Constructores y métodos sobrecargados.</li> <li><input type="radio"/> Métodos accesorios o auxiliares</li> <li>• Documentación de clases y de métodos</li> <li>• Constantes y variables de clase.</li> <li>• Interfaz e implementación de una clase</li> <li>• Librería de clases del lenguaje</li> <li><input type="radio"/> Clases de fines específicos</li> <li>• Agrupamiento de objetos.</li> <li><input type="radio"/> Colecciones de tamaño fijo y de tamaño variable.</li> <li>• Manejo de errores.</li> <li>• Noción de evento.</li> <li>• Interfaz de usuario.</li> </ul>
<p>Base de Datos</p>	<p>Analizar datos y su relación con la aplicación a desarrollar pudiendo determinar cuáles son necesarios a utilizarse según el momento dado.</p> <p>Administrar una base de datos utilizando una DBMS, configurando permisos, creando bases de datos, configurando copias de seguridad, utilizando DDL.</p> <p>Interpretar Queries por medio del estándar SQL, realizando distintos tipos de consultas a la base de datos.</p>	<p>Organización Lógica de los datos</p>	<ul style="list-style-type: none"> <li>• Concepto de datos, procesos, salida.</li> <li>• Noción de registro y Concepto de archivo.</li> <li>• Noción de base de datos.</li> <li>• Ventajas de las bases de datos.</li> <li>• Tipos de bases de datos.</li> <li><input type="radio"/> Base de datos relacionales.</li> <li><input type="radio"/> Base de datos NoSQL.</li> <li>• Concepto de Tabla.</li> <li>• Concepto de:</li> <li><input type="radio"/> Entidad.</li> <li><input type="radio"/> Relaciones entre entidades.</li> <li><input type="radio"/> Atributo de las entidades.</li> <li><input type="radio"/> Clave principal.</li> <li><input type="radio"/> Clave secundaria.</li> <li><input type="radio"/> Cardinalidad de las relaciones.</li> <li>• Modelo Entidad-Relación.</li> <li><input type="radio"/> Diagrama Entidad-Relación (ER).</li> <li>• Concepto de normalización y redundancia.</li> <li>• Consultas anidadas.</li> <li>• Tablas relacionadas.</li> <li>• Consultas de datos.</li> </ul>

*[Handwritten signature]*



			<ul style="list-style-type: none"> <li>• Altas, Bajas y Modificaciones (ABM).</li> </ul>
		Manipulación de base de datos.	<ul style="list-style-type: none"> <li>• Lenguaje de Consulta</li> <li>○ Software libre.</li> <li>○ Software propietario.</li> <li>• Construcción de Base de datos.</li> <li>• Consultas sobre una tabla.</li> <li>• Selección simple.</li> <li>• Eliminación de respuestas duplicadas.</li> <li>• Selección ordenada.</li> <li>• Selección condicionada.</li> <li>• Operadores de comparación.</li> <li>• Operadores lógicos.</li> <li>• Campos calculados.</li> <li>• Funciones agregadas de dominio (mínimo, máximo, suma, promedio).</li> <li>• Selección con agrupamientos (agrupar).</li> <li>• Altas, bajas y modificaciones (ABM).</li> <li>• Actualización de un registro.</li> <li>• Actualización de registros múltiples</li> <li>• Actualización condicionada.</li> <li>• Inserción y eliminación de un registro.</li> <li>• Eliminación de múltiples registros.</li> <li>• ORM.</li> </ul>

**ESTRATEGIAS DE ENSEÑANZA:**

El presente diseño se enmarca en el enfoque de formación por capacidades profesionales-competencias que responde a la necesidad de encontrar un punto de convergencia entre educación y trabajo, que tiene como eje una formación de calidad vinculada con las necesidades de los individuos y con las necesidades del sector productivo. Este se presenta como una alternativa para responder desde el punto de vista formativo a las necesidades derivadas

de las transformaciones del mundo del trabajo.

Como toda formación profesional basada en el enfoque de competencias laborales intenta promover el aprendizaje, práctico, integral y activo. Busca como resultado cambios en la forma de actuar de las personas. En este sentido, se destaca el valor de la experiencia (propia y de otros) que adquieren significación a partir del acercamiento a situaciones vinculadas con la realidad del trabajo en el sector y del aporte de compañeros y compañeras más experimentados/as.

El enfoque de competencias también plantea particularidades en la definición de metodología de aprendizaje. Por tratarse de un enfoque que privilegia la experiencia, los conocimientos y su puesta en práctica en el ámbito laboral, se estructura en la metodología de resolución de problemas. Esta implica un proceso de indagación hacia la resolución de preguntas sobre situaciones complejas.

Implica recabar información para procesarla, aplicarla y resolver cualquier situación y por lo tanto permite la integración de contenidos. Promueve una postura activa. Estimula el pensamiento crítico y reflexivo. Favorece el intercambio con lxs otrxs.

Este tipo de aprendizaje basado en problemas, promueve la interacción grupal considerándola uno de los ejes en la metodología de aprendizaje. La realización de trabajos en equipo y el intercambio con otros las estimula el aprendizaje de los conocimientos y experiencia del otro /a, la capacidad de escucha. Implica también aprender a defender las propias ideas, genera vínculos afectivos que favorecen el hecho de aprender, estimula la pertenencia a un grupo. Por otro lado, cuando grupalmente se propone la resolución de problemas, el aporte de cada persona colabora en la construcción de estrategias creativas que aprovechan la sinergia grupal para potenciar la producción del aprendizaje.

La organización curricular propuesta considera la articulación entre práctica y teoría a través del diseño de módulos, centrados en torno a situaciones problemáticas derivadas del contexto laboral y expresado para su tratamiento en proyectos, simulaciones, análisis de casos, actividades de mejora continua o búsqueda de soluciones apropiadas, entre otras alternativas.

vas.



El énfasis estará puesto en todo momento en el desarrollo de las capacidades, no a los contenidos como fines en sí mismos. Estos pueden ampliarse a aquellos demandados por la situación problemática que se esté resolviendo<sup>2</sup>.

Cuando hablamos de resolución de situaciones problemáticas hacemos referencia a aquellas que reflejen las que habitualmente deberá afrontar el carpintero y armador en su actividad laboral. Para ello resulta conveniente recurrir a la simulación de situaciones como: la observación de campo y el análisis de casos que permitan adoptar las técnicas Diseño Curricular Armador y montador de tabiques y cielorrasos de placas de roca de yeso de trabajo según las reglas del arte del oficio, las disposiciones de higiene y seguridad, las relaciones interpersonales, la calidad del trabajo, a modo de vincular la acción con la reflexión sobre la propia práctica. Desde el punto de vista de la organización de la clase, se propone combinar actividades individuales y grupales.

Las actividades individuales generalmente se utilizan cuando se requiere desarrollar competencias en profundidad o realizar síntesis de conocimientos.

Las actividades grupales pueden generarse en grupos pequeños o en debate plenario; se utilizan en general cuando se demanda comprensión, análisis y reflexión sobre la práctica y sus fundamentos, producción y propuestas de mejoras, entre otras capacidades.

Se sugiere planificar actividades formativas en función de las características de los participantes para coadyuvar al logro de los aprendizajes, adecuando el avance al desarrollo de cada persona y del propio grupo.

Las actividades se organizan, considerando los distintos momentos en la secuencia didáctica, en iniciales, de desarrollo y de cierre o finales.

Actividades de inicio: Se relacionan con todo el módulo. Su propósito es explorar los saberes previos, las expectativas e intereses de los cursantes; presentar de manera significativa el módulo; plantear la situación problemática y explicitar los objetivos y la metodología de trabajo para que los participantes sepan qué van a realizar, por qué se propone y cómo van a

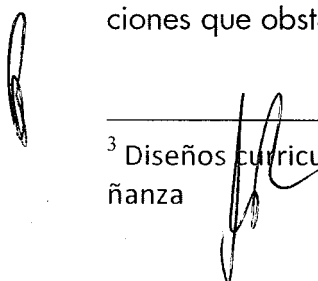
  
<sup>2</sup> Diseño Curricular .Marco Pedagógico. Ministerio de Trabajo Empleo y Seguridad Social.

trabajar. Los saberes y experiencias previas de los participantes constituyen la base para promover la construcción de aprendizajes significativos, por ello todo nuevo proceso de enseñanza debe comenzar con una evaluación de dichos saberes en función de los objetivos propuestos.

Actividades de desarrollo: Se determinan fundamentalmente por el núcleo de capacidades que se pretende desarrollar. Pueden hacer referencia a todo el módulo, o en su defecto a cada unidad o bloque, si se han incluido. En esta instancia se sugiere a partir de la información diagnóstica, promover la participación activa de los participantes en la construcción de sus procesos de aprendizaje y establecer la mayor cantidad de relaciones en el material que se pretende enseñar con la finalidad de ayudar a que el participante lo asimile a sus esquemas previos y pueda reestructurar sus saberes en niveles crecientes de complejidad. En este marco, el error se valoriza como un medio para comprender el proceso de aprendizaje del cursante y una ayuda para superarlo. En el desarrollo resulta necesario integrar las dimensiones de la competencia e interrelacionar en forma equilibrada las actividades prácticas con las de reflexión, las ejemplificaciones, los debates, las explicaciones y las demostraciones adecuadamente, de modo de contribuir al logro de las capacidades requeridas.

Actividades de cierre: Se relacionan con todo el módulo. Su propósito es promover la integración y aplicación del aprendizaje (realización de una tarea, elaboración de un producto, resolución del problema). Esta actividad, si bien es la culminación del proceso de aprendizaje del módulo se puede comenzar a elaborar desde el comienzo ya que puede constituir un punto de referencia para planificar las actividades de desarrollo. En un primer momento, nos podemos interrogar ¿qué situación vamos a plantear para que los participantes puedan evidenciar el aprendizaje logrado?<sup>3</sup>

Serán pilares centrales de todas las estrategias de enseñanza la interacción grupal, el respeto por los ritmos del aprendizaje y perspectivas de cada uno y el acompañamiento ante situaciones que obstaculicen la trayectoria de formación del estudiante, ya sea en su permanencia

  
<sup>3</sup> Diseños curriculares. Ministerio de Trabajo, Empleo y Seguridad Social. Criterios para la enseñanza

o en la apropiación de contenidos. El rol del tutor, que trabaja junto al docente, es central en relación con estos aspectos.

La planificación de actividades formativas alternativas, formuladas a partir del conocimiento de la heterogeneidad de los cursantes, el seguimiento de la asistencia y el monitoreo de las dificultades y el consecuente acompañamiento ante las dificultades son claves dentro de la tarea del tutor.<sup>1</sup>

#### Prácticas profesionalizantes

Dice en todos los ítems lo mismo. Habría que unificarlo en un único párrafo y especificar si las prácticas se realizarán durante la misma cursada...

#### Técnicas de Programación

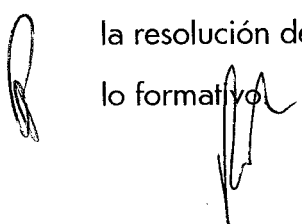
En relación a las prácticas formativas de carácter profesionalizante, son un eje estratégico de la propuesta pedagógica para el ámbito de la FP, al situar al participante en los ámbitos reales de trabajo con las problemáticas características que efectivamente surgen en la planificación de procedimientos o secuencias de actividades asociada al desarrollo de algoritmos y la resolución de problemas de base computacional, y que se organiza a nivel de cada módulo formativo.

#### Ingeniería de Software, Relaciones Laborales y Orientación Profesional

En relación a las prácticas formativas de carácter profesionalizante, se definen como unos de los ejes estratégicos de la propuesta pedagógica para el ámbito de la FP, el situar al participante en los ámbitos reales de trabajo con las problemáticas características de desempeño ocupacional/profesional.

#### Programación Orientada a Objetos

En relación a las prácticas formativas de carácter profesionalizante, son un eje estratégico de la propuesta pedagógica para el ámbito de la FP, al situar al participante en los ámbitos reales de trabajo con las problemáticas características que efectivamente surgen en la planificación de procedimientos o secuencias de actividades asociada al desarrollo de algoritmos y la resolución de problemas de base computacional, y que se organiza a nivel de cada módulo formativo.



## Base de Datos

En relación a las prácticas formativas de carácter profesionalizante, se definen como unos de los ejes estratégicos de la propuesta pedagógica para el ámbito de la FP, el situar al participante en los ámbitos reales de trabajo con las problemáticas características que efectivamente surgen en la planificación de procedimientos o secuencias de actividades asociada al uso y manipulación de los datos, del lenguaje de consulta y la resolución de problemas relativos al flujo de datos dentro de los problemas de tipo computacional.

## MATERIALES DIDÁCTICOS:

Para el dictado del curso se contará con aulas informatizadas con 14 equipos con el software correspondiente. (Java, Netbeans, Mysql, Dia, etc.), pizarrón, marcadores y con presentaciones del curso.

## MECANISMOS DE EVALUACIÓN:

La evaluación y acreditación se realiza por módulos en forma independiente. Asimismo, se otorga importancia a la evaluación de los procesos realizada en forma continua, como resultado de la observación y la reflexión de la actividad habitual, con el propósito de mejorar la enseñanza y el aprendizaje.

Es importante desde este enfoque el desarrollo de capacidades de autoevaluación, como parte del proceso de apropiación de los contenidos. Y que esto hace referencia a que el sujeto pueda analizar en profundidad las situaciones de aprendizaje y vincularlas a su entorno laboral.

De ese modo logra:

- El desarrollo de capacidades de auto referencia y autorregulación del aprendizaje
- La estimulación de la apropiación de la actividad de conocer como desempeño individual.
- La responsabilidad en la apropiación de los contenidos
- La identificación de formador/a como acompañante de los logros personales y gru-

pales.

Esto se plantea no sólo como estrategia metodológica sino también como herramientas para el desarrollo de capacidades hacia la práctica profesional enmarcada en procesos productivos de calidad y seguridad en el ámbito laboral.

Un aspecto importante dentro de este enfoque lo constituye el tipo de evaluación empleada durante el proceso de aprendizaje.

Se proponen tres tipos de evaluaciones:

- Evaluación diagnóstica: se efectúa al iniciar cada unidad de aprendizaje. Constituye un instrumento que permite tener en cuenta las capacidades adquiridas por los/as participantes a través de sus experiencias anteriores (formativas, laborales), los conocimientos y las actitudes que evidencian ante los distintos procesos de producción, el modo de interacción que se produce en las actividades grupales.
- Evaluación del proceso de trabajo individual y grupal: será continua. A partir de las propuestas de actividades individuales y grupales en la puesta en práctica de los módulos de formación. Considerará el desarrollo de cada módulo diferenciando momentos de inicio, desarrollo y cierre de cada módulo. De acuerdo con el desarrollo de estas actividades, habrá momentos de intercambio y evaluación individual y grupal. Se estimula también el desarrollo de actividades de autoevaluación buscando que cada participante desarrolle estrategias metacognitivas para identificar: sus expectativas frente al curso, reconocer sus conocimientos previos, evaluar su desempeño en el trabajo grupal y reconocer sus logros en el aprendizaje.
- Evaluación final o certificación: se ajustará a las capacidades que se espera lograr al finalizar cada módulo. Pueden presentarse una instancia individual y otra grupal.

Se lleva a cabo a partir de tres tipos de instrumentos:

- Lista de cotejo
- Ejercicio de integración
- Evaluación oral y/o escrita de la unidad de aprendizaje

La instancia de evaluación individual considera sobre todo el grado de acercamiento a la problemática planteada en el módulo, de apropiación de los contenidos y de aplicación a las

situaciones de trabajo cotidianas. Esta instancia será de modo oral o escrita, buscando sobre todo la aplicación de los contenidos a su situación de trabajo.

La instancia de evaluación grupal tendrá en cuenta también el grado de apropiación y aplicación de los contenidos como la dinámica grupal de resolución de problemas en conjunto o elaboración de productos y el análisis de incidentes ocurridos en el proceso productivo, tendientes a generar mejores respuesta frente a las condiciones de trabajo específicas de su sector.

#### CRONOGRAMA DE TRABAJO:

El docente ubicará los módulos y niveles en un cuadro como el que se presenta a continuación:

Módulo	Bloque	Semanas	Encuentros
Técnicas de Programación	Elementos Informáticos	2 semanas	4 encuentros
	Diseño de Algoritmos	5 semanas	10 encuentros
	Desarrollo de Programas	5 semanas	10 encuentros
Subtotal		12 semanas	24 encuentros
Ingeniería de Software, Relaciones Laborales y Orientación Profesional	Ingeniería de Software	3 semanas	6 encuentros
	Relaciones Laborales	1 semana	2 encuentros
	Orientación Profesional y Formativa	1 semana	2 encuentros
Subtotal		5 semanas	10 encuentros
Programación Orientada a Objetos	Fundamentos de la Programación Orientada a Objetos.	5 semanas	10 encuentros
	Metodología de Desarrollo.	5 semanas	10 encuentros
	Lenguaje de Programación Orientado a Objetos.	5 semanas	10 encuentros
Subtotal		15 semanas	30 encuentros
Base de datos	Organización Lógica de los datos	3 semanas	6 encuentros
	Manipulación de base de datos	4 semanas	8 encuentros
Subtotal		7 semanas	14 encuentros
Total		39 semanas	78 encuentros

ANEXO ÚNICO RESOLUCIÓN (CS) N°: 025-19