

## **1. DENOMINACIÓN DE LA CARRERA**

# **TECNICATURA SUPERIOR EN DESARROLLO DE SOFTWARE**

**1.1. Nivel: SUPERIOR**

**1.2. Acreditación:**

**Se otorga el título de**

---

**TÉCNICO SUPERIOR**

**EN DESARROLLO DE SOFTWARE**

---

## 1. IDENTIFICACIÓN DE LA PROPUESTA

a. Normativa jurisdiccional de aprobación:	Res. Min. Nro.
b. Sector/es de actividad socio productiva	Informática (Software y servicios informáticos)
c. Denominación del perfil profesional	Desarrollador de Software
d. Familia profesional	Informática
e. Denominación del título	Técnico Superior en Desarrollo de Software
f. Nivel y ámbito de la trayectoria formativa/carrera:	Nivel Superior en la modalidad técnico profesional
g. Duración de la trayectoria formativa:	3 años
h. Cohortes para las que se pide la validez de la titulación (desde-hasta)	Según corresponda cuando se implemente

## 2. FUNDAMENTACIÓN

La industria de la informática en Argentina y en el mundo está en un momento de apogeo, en el cual los servicios informáticos y profesionales que se producen no sólo se consumen en nuestro país sino que se exportan a diferentes países generando la demanda de miles de puestos de trabajo genuinos que generan las condiciones para un continuo crecimiento y perfeccionamiento. Esto implica que las condiciones del mercado donde se incorpora este desarrollo tecnológico y profesional son de carácter universal y se encuentra definido por reglamentaciones profesionales y laborales. Para todas las organizaciones con o sin fines de lucro, públicas o privadas, la necesidad de software específicos se ha convertido en una oportunidad transversal, generando una demanda creciente que en la actualidad no muestra un techo de satisfacción. Éstas necesitan cada vez más sistemas que apoyen a la gestión industrial, productiva y administrativa a fin de contar con información clara y precisa que ayude en la toma de decisiones para generar procesos y negocios más eficientes y competitivos.

En cuanto al ámbito Provincial y Nacional también se manifiesta una creciente demanda de profesionales con la modalidad "homeworking" (trabajo desde el hogar); que coexiste con el trabajo deslocalizado en otros países de la región y de otros continentes.

Acompañando esta realidad se sanciona la Ley de Economía del Conocimiento (LEC) N° 27.570 con el objetivo de promover nuevas tecnologías, generar valor agregado, fomentar el empleo de calidad y facilitar el desarrollo de empresas que se dediquen a la industria del software, entre otras. Por ello, Argentina se encuentra en un contexto propicio para este crecimiento: *"ampliar el alcance de los beneficios para las empresas del sector incluyendo un mayor número de las mismas haciendo crecer el ecosistema y sus oportunidades."*

Asimismo, en la provincia de Córdoba, el ejercicio de las profesiones en ciencias informáticas está regulado por la Ley N° 7642/87 indicando en el artículo 2 el

reconocimiento a profesionales que posean títulos oficiales con validez a nivel nacional o provincial que se alcanzan con tres o más años lectivos de estudio.

Esta Tecnicatura Superior se encuadra en los marcos regulatorios, normativas y políticas educativas nacionales y provinciales, en busca de integrarse en un proyecto social en el que la educación cobra un papel relevante. Estas premisas orientan los esfuerzos y desafíos que asume la propuesta de formación y recupera también la necesidad de dar respuestas a las demandas concretas que se generan desde los sectores socio económicos y productivos.

### **3. OBJETIVO DE LA CARRERA**

Formar a un técnico que sea capaz de adaptarse a la intervención y desarrollo en el proceso del ciclo de vida de un software, donde las principales actividades son las de diseñar, construir y verificar artefactos de software correspondientes a sistemas de información o vinculados directamente al hardware o sistemas de comunicación de datos. Este Técnico Superior en Desarrollo de Software podrá desempeñarse en empresas o como Freelancer, cubriendo así las demandas globales de esta área. Adquirirá también las capacidades emprendedoras requeridas para ser protagonistas de procesos de cambio, dirigidos a mejorar la empleabilidad, la productividad y la construcción de sus propios proyectos ocupacionales.

### **4. REQUISITOS DE INGRESO**

Haber cumplimentado la educación obligatoria en cualquiera de sus formas y/o estar comprendido en las condiciones que fija la Resolución Ministerial 412/10.

### **5. PERFIL PROFESIONAL DEL TÉCNICO SUPERIOR EN DESARROLLO DE SOFTWARE**

El Técnico Superior en Desarrollo de Software se ocupa de relevar, diagnosticar, diseñar, programar, testear, implementar y dar soporte a problemáticas organizacionales a través de soluciones tecnológicas, mediante artefactos de software de diferentes plataformas, interfaces y dispositivos, trabajando tanto en relación de dependencia como emprendiendo y comercializando sus propios desarrollos bajo normas de calidad y en la modalidad de teletrabajo o trabajo remoto.

Es un profesional con una fuerte cultura digital activa y capacidad de trabajo en equipo, que le permite detectar soluciones de base tecnológica a problemáticas de índole cotidiano en los diversos procesos empresariales.

Es capaz de adaptarse a distintos modelos de negocio, interpretando la dinámica de cada sector y llevando a cabo soluciones en cada proceso que amerite un cambio radical o de mejora continua. Está calificado para colaborar y participar en los diferentes estudios de factibilidad tanto económica, técnica u operativa, como así también en la evaluación y selección de los equipos de procesamiento, comunicación y almacenamiento.

El Técnico Superior en Desarrollo de Software realiza prácticas interdisciplinarias para proyectos de emprendimiento en la sociedad, teniendo en cuenta que su actividad, enmarcada dentro de la industria del software, aporta a la economía del conocimiento generando valor a través de nuevos productos y servicios.

Para poder desarrollar plenamente su profesionalidad, el técnico posee una fuerte vinculación con los aspectos deontológicos de la misma y con los éticos de la vida en sociedad, en el marco de un Estado democrático. En aquellos aspectos vinculados con la construcción de conocimiento, su visualización y comprensión de modelos de apoyo a la toma de decisiones. Contribuye a la resolución de problemas mediante una comunicación eficaz fundada en el respeto y la valoración del tiempo.

## **6. ALCANCE DEL TÉCNICO SUPERIOR EN DESARROLLO DE SOFTWARE**

El Técnico Superior en Desarrollo de Software estará capacitado para producir componentes de software, lo que comprende su diseño detallado, construcción y realización de la verificación unitaria de los mismos así como su depuración, optimización y mantenimiento, desarrollando las actividades descritas en el perfil.

Se establece como alcance:

- modelizar y construir aplicaciones de software a partir de especificaciones
- verificar la calidad de las aplicaciones, diseñando casos de prueba unitarios y de integración
- optimizar y depurar código de aplicaciones de software para resolver defectos o mejorarlos
- documentar en base a normas o estándares establecidos
- gestionar sus propias actividades dentro del equipo de trabajo del proyecto
- determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua.

## **7. FUNCIONES QUE EJERCE EL PROFESIONAL**

Del análisis de las actividades profesionales que se desprenden del Perfil Profesional se establecen como funciones del Técnico Superior en Desarrollo de Software:

- **Modelizar y construir aplicaciones de software a partir de especificaciones**
  - Interpretar críticamente las especificaciones recibidas.
  - Interpretar la arquitectura del sistema en el cual se inserta la asignación.
  - Aplicar patrones de diseño si corresponde.
  - Diseñar la solución.
  - Representar el diseño.
  - Verificar el diseño.
  - Reutilizar elementos ya existentes.
  - Redactar código.
  - Testear el código
  - Optimizar el código.
  - Controlar cambios y versiones.
  - Utilizar ambientes de desarrollo.
  - Valorar opciones de capacitación continua, tanto propias como del equipo de trabajo.

- **Verificar la calidad de las aplicaciones, diseñando casos de prueba unitarios y de integración**

esto comprende:

- Considerar las necesidades de cobertura de la prueba.
- Diseñar los casos de prueba.
- Preparar el entorno de pruebas.
- Realizar pruebas unitarias.
- Registrar casos de prueba, datos y resultados de pruebas y acciones correctivas.

- **Optimizar y depurar código de aplicaciones de software para resolver defectos o mejorarlos**

esto comprende:

- Interpretar código.
- Diagnosticar defectos.
- Documentar defectos e Informarlos
- Depurar defectos.

- **Documentar en base a normas o estándares establecidos**

esto comprende:

esto comprende:

- Registrar actividades realizadas.
- Documentar todos los productos de su labor con los requerimientos funcionales y técnicos y de normas de calidad.

- **Gestionar sus propias actividades dentro del equipo de trabajo del proyecto**

esto comprende:

- Obtener métricas a partir de los registros de actividades.
- Reportar avances y dificultades.
- Planificar sus actividades.
- Controlar sus actividades.
- Coordinar actividades de acuerdo a roles y funciones del equipo de trabajo.
- Elaborar interrogantes coherentes al problema o proyecto planteado
- Escuchar y reformular conceptos en interrelación dialogada.
- Recibir indicaciones
- Reconocer sus potencialidades personales y ajenas al momento de integrar equipos de trabajo
- Jerarquizar metas y determinar secuencias y plazos de realización

- **Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua**

esto comprende:

- Aprender nuevos lenguajes y tecnologías.

- Apropiarse de su actualización personal en las tendencias y demandas del mercado
- Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

## **8. ÁREA OCUPACIONAL**

Este técnico puede trabajar en organizaciones de diversos tipos; empresas que realizan desarrollo de software por encargo de organizaciones locales o extranjeras, que proveen software junto con otros servicios de asesoramiento y consultoría, y, en menor número, que desarrollan sus propios productos de software para vender en el país o en el exterior. También en organizaciones dedicadas a otros servicios, pero que producen el software que necesitan para desarrollar sus propias actividades o que integran en productos que venden. Su posición ocupacional suele denominarse analista programador, programador, o desarrollador de software. Integra equipos de proyecto dedicados al desarrollo o mantenimiento de software y recibe asignaciones específicas que tiene que resolver en lapsos que suelen medirse en términos de días o semanas, produciendo artefactos que satisfagan especificaciones y se integren al sistema objeto del proyecto.

Resuelve estas asignaciones individualmente o trabajando en pares, recibiendo la supervisión y asesoramiento de un líder de proyecto o de grupo, con quien consulta dudas y decisiones significativas o comunica inconvenientes. También recibe apoyo y brinda colaboración a otros miembros del grupo. Su trabajo es verificado por un grupo de “testing” y eventuales controles cruzados de código importante. Con una mayor experiencia o especialización en determinadas tecnologías o metodologías, posibles evoluciones ocupacionales del Técnico Superior en Desarrollo de Software son el liderar grupos de trabajo o asumir roles de analista técnico en la materia de su especialidad.

Asimismo, 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. Por otra parte, Técnicos Superiores en Desarrollo de Software o profesionales equivalentes con capacidad emprendedora pueden y suelen asociarse entre ellos para generar sus propias empresas para brindar servicios de desarrollo y proveer software a terceros.

## **9. HABILITACIONES PROFESIONALES**

Las actividades que realiza y para las cuales está capacitado el Técnico Superior en Desarrollo de Software, así como el ámbito de su desempeño y el campo y condiciones de su ejercicio profesional son los descritos en el Perfil Profesional correspondiente.

Si bien las actividades de este técnico superior no están orientadas a un tipo de software en particular, conviene tener en cuenta que el software es utilizado crecientemente en sistemas que afectan a la seguridad pública. Estos sistemas, denominados críticos para la seguridad, son lo que, en un sentido general, involucran riesgos que conllevan la posibilidad de pérdidas inaceptables (daños para la salud o aún la vida humana, daños a la propiedad, contaminación ambiental, conflictos sociales, grandes pérdidas monetarias).

En función de estos riesgos, se establecen las siguientes habilitaciones profesionales para el Técnico Superior en Desarrollo de Software, con las limitaciones o exclusiones que se

indican en cada caso. Estas habilitaciones tienen efecto para su desempeño en forma autónoma o asumiendo plenamente la responsabilidad por los resultados que obtenga su grupo de trabajo.

- Diseñar, construir y verificar artefactos de software de complejidad media, correspondientes a sistemas de información o vinculados indirectamente al hardware o a sistemas de comunicación de datos, respondiendo a especificaciones.

Queda excluido de esta habilitación el software correspondiente a sistemas críticos para la seguridad, como es el caso de los que involucren el procesamiento de información que conlleve riesgos efectivos para terceros. Particularmente, queda excluido el software destinado a:

- control de equipos y procesos médicos, industriales o de domótica que puedan poner en riesgo inmediato o mediano la salud de personas;
- procesamiento de información crítica para los individuos, como ser la que sirva para corroborar su identidad o características de su estado de salud, para demostrar situaciones legal, fiscal, patrimonial u otras que afecten a su patrimonio o a sus libertades;
- procesamiento en línea de transacciones financieras importantes.

En estos casos, requerirá la supervisión de profesionales habilitados.

- Controlar la calidad de artefactos de software para resolver defectos o mejorarlos, lo que incluye revisar especificaciones, diseños y código.

Esto se realiza bajo supervisión en el marco de equipos de desarrollo de software.

## 10. DISEÑO Y ORGANIZACIÓN CURRICULAR - Plan de Estudios

### 10.1. RÉGIMEN DE CURSADA Y CARGA HORARIA

Duración de la carrera: 3 años  
 Modalidad de cursado: presencial  
 Carga horaria total: **1623 horas reloj.**  
 Régimen de cursado: Anual/ Cuatrimestral

### 10.2. ESTRUCTURA CURRICULAR

Nro de orden	Campo de formación	Forma to	Primer Año	Hs. Cat semanales	Hs. reloj anuales	Correlati -vas
			Anuales			
1	FF	A	Elementos de matemática y lógica	3	64	-
2	FF	A	Sistemas y organizaciones	4	85	-
3	FE	A	Programación I	5	107	-
4	FE	A	Base de datos	4	85	-
<b>Cuatrimestrales</b>				Hs. Cat semanales	Hs. reloj cuatrim	Correlati -vas
<b>1er. Cuatrimestre</b>						
5	FG	MT	Competencias Comunicacionales I	3	32	-
6	FG	MT	Aproximación al mundo del trabajo	3	32	-
<b>2do. Cuatrimestre</b>						

7	FF	MA	Arquitectura de las computadoras	4	43	-
8	FG	MT	Competencias Comunicacionales II	3	32	5
9	FG	MT	Ética y deontología profesional	3	32	6
<b>Segundo Año</b>						
<b>Anuales</b>				Hs. Cat semanales	Hs. reloj anuales	Correlati -vas
10	FF	A	Inglés	4	85	-
11	FF	A	Estadística y probabilidad aplicadas	3	64	1
12	FE	A	Modelado y Arquitectura de Software	4	85	2
13	FE	A	Programación II	5	107	3, 4
14	PP	T	Práctica Profesionalizante I	7	149	2, 3, 8 y 9
<b>Cuatrimestrales</b>						
<b>1er. Cuatrimestre</b>				Hs. Cat semanales	Hs. reloj cuatrim	Correlati -vas
15	FF	MA	Sistemas operativos	4	43	7
<b>2do. Cuatrimestre</b>						
16	FF	MA	Redes	4	43	15
<b>Tercer Año</b>						
<b>Anuales</b>				Hs. Cat semanales	Hs. reloj anuales	Correlati -vas
17	FE	A	Interfaz de usuario	3	64	2
18	FE	A	Ingeniería de software	5	107	12
19	FE	A	Programación III	5	107	13, 15, 16
20	PP	T	Práctica Profesionalizante II	8	171	de 10,a 14
<b>Cuatrimestrales</b>						
<b>1er. Cuatrimestre</b>				Hs. Cat semanales	Hs. reloj cuatrim	Correlati -vas
21	FG	MT	Gestión de proyectos	4	43	14
<b>2do. Cuatrimestre</b>						
22	FE	A	Verificación y Validación de programas	4	43	13
			<b>TOTAL DE HORAS RELOJ</b>		<b>1623</b>	

### Formato de los espacios curriculares:

**A: Asignatura:** se plantea alrededor de un campo disciplinar, se promueve la comprensión abarcativa e integradora de enfoques, teorías y problemas en debate dentro de ese campo, categorías y saberes de referencia articulados, para andamiar conceptualmente el análisis integrador de la realidad dentro de múltiples contextos.

**M: Módulo:** Es una unidad que integra contenidos de diversa naturaleza que se organizan en base a la resolución de situaciones problemáticas propias del campo profesional. Esos problemas o cuestiones son ejes que contribuyen a articular, dar coherencia e integrar los contenidos de enseñanza en prácticas que reproducen o modelizan condiciones y exigencias propias del ejercicio profesional del perfil de referencia.



Es por esto que los módulos articulan e integran los saberes con problemas relevantes del campo ocupacional al que se refiere y requieren la combinación de la teoría y la práctica, la acción y la reflexión.

De este modo, saber y saber hacer, son y se presentan como complementarios en la tarea de acción/reflexión/comprensión en el proceso de formación

**T: Taller:** pretende el aprendizaje por entrenamiento y su principal característica es la participación de los estudiantes orientada a la resolución colectiva de situaciones problemáticas. Enfatiza la íntima relación entre los saberes de referencia y los saberes prácticos, articulando los fundamentos conceptuales, las experiencias previas y la propia acción.

### 10.3. CAPACIDADES Y CONTENIDOS MÍNIMOS<sup>1</sup>

#### PRIMER AÑO

#### 1. Elementos de matemática y lógica

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

Interpretar críticamente las especificaciones recibidas.

Diseñar la solución.

Representar el diseño.

Redactar código.

Testear el código

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Aplicar métodos inductivos, deductivos y recursivos en la resolución de situaciones problemáticas y demostraciones matemáticas.</li> <li>● Efectuar cálculos en teoría de números, para su posterior utilización en diversas áreas del desarrollo de software.</li> <li>● Aplicar los conocimientos adquiridos para evaluar la viabilidad de las soluciones propuestas desde un enfoque lógico formal.</li> <li>● Aplicar la optimización de cierto sistema o software desde modelos lógicos formales (complejidad temporal).</li> </ul>	<p>Funciones; tipos: Inyectivas, sobreyectivas, inversas, composición. Relaciones: tipos: reflexividad, simetría, transitividad, equivalencia. Conjuntos; diagrama de Venn, operaciones, complementos, producto cartesiano, conjunto potencia. Numerabilidad y cardinalidad. Aritmética modular.</p> <p>Relaciones de congruencia. Sistemas de numeración.</p> <p>Elementos de lógica. Lógica proposicional, conectivos lógicos. Tablas de verdad. Formas normales; conjuntiva y disyuntiva. Validez. Lógica de predicados; cuantificadores universal y existencial.</p>

<sup>1</sup> Las tecnologías, plataformas y lenguajes mencionados son prioritarios sólo pueden ser reemplazados por otros de similares características

<ul style="list-style-type: none"> <li>● Justificar resultados particulares obtenidos con propiedades generales demostradas, para fomentar el pensamiento matemático y crítico, usando la teoría elemental de números.</li> <li>● Determinar familias de conjuntos (conjunto potencia, partición de un conjunto) y el producto cartesiano de conjuntos, para su posterior utilización en relaciones binarias, de utilidad en el resto de las asignaturas del área de programación.</li> <li>● Interpretar el formalismo de conjuntos, para poder especificar colecciones formalmente y operar con ellas, tanto en la matemática como en las ciencias de cómputo.</li> <li>● Adquirir habilidades teóricas y prácticas para generar productos solicitados aplicando los sistemas numéricos y la geometría en el desarrollo de aplicaciones.</li> <li>● Interpretar el formalismo lógico, para traducir enunciados del lenguaje natural al lenguaje formal, capacidad necesaria en la especificación de requerimientos y formulación de problemas.</li> <li>● Aplicar las propiedades de las operaciones lógicas, con el fin de identificar equivalencias y relaciones de consecuencia lógica, que se aplican en los razonamientos deductivos y las demostraciones de teoremas.</li> </ul>	<p>Modus ponens y modus tollens. Limitaciones de la lógica de predicados. Técnicas de demostración. Nociones de implicación, conversa, inversa, contrapositivo, negación y contradicción. La estructura de las demostraciones matemáticas. Demostración directa por contraejemplo, por contradicción. Inducción matemática. Inducción fuerte. Definiciones matemáticas recursivas. Buen ordenamiento.</p> <p>Bases de conteo. Argumentos de conteo, regla de la suma y el producto. Principio de inclusión-exclusión. Sucesiones aritméticas y geométricas números de Fibonacci. Principio de Dirichlet. Números reales. Funciones reales de una variable. Límites y continuidad. Cálculo diferencial. Álgebra lineal, vectores y matrices.</p>
--	---

## **2. Sistemas y Organizaciones**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto:

esto comprende:

Escuchar y reformular conceptos en interrelación dialogada.

Elaborar interrogantes coherentes al problema o proyecto planteado

Recibir indicaciones

Jerarquizar metas y determinar secuencias y plazos de realización

Planificar sus actividades.

## Controlar actividades

- Documentar en base a normas o estándares establecidos esto comprende:

Registrar actividades realizadas.

Documentar todos los productos de su labor con los requerimientos funcionales y técnicos y de normas de calidad.

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"><li>● Reconocer los aportes fundamentales de la Teoría General de Sistemas y del Enfoque Sistémico.</li><li>● Reconocer los procesos principales de las organizaciones comerciales y de servicios para identificar objetivos, recursos y actividades considerando los modelos básicos de representación.</li><li>● Identificar los documentos, formularios y personas intervinientes en los procesos de una organización para conformar el modelo que permita, luego, asignar un sistema de información.</li><li>● Reconocer las funciones administrativas de la Organización para reconocer las organizaciones como espacio de demanda de tecnología de la información.</li><li>● Reconocer las necesidades de información presentes en distintos contextos para describir, en forma simple, el sistema de información que da respuesta.</li><li>● Aplicar herramientas de planeamiento para la elaboración de planes simples considerando objetivos, recursos y restricciones en casos reales enmaquetados.</li></ul>	<p>Elementos de teoría general de los sistemas, enfoque sistémico de la organización. Elementos de estructura y comportamiento de las organizaciones, organización estructurada por funciones o líneas de productos, el manejo de sedes. Concepto de proceso. Procesos del ciclo de ventas y cobranzas; del ciclo de compras y pagos. Nociones de procesos de gestión y transformación de materiales y su organización. Comprobantes usuales, requerimientos legales y fiscales. Concepto de recurso y su gestión. El papel de los sistemas de información en la organización. Nociones de control interno. La contabilidad como sistema de información. Algunas características de organizaciones y procesos de servicios.</p> <p>Los niveles de la administración: la planificación estratégica, el control de gestión, el control operativo y el tipo de sistemas de información asociados a los mismos.</p> <p>Conceptos de Tecnología de la Información, evolución histórica, tecnologías que la integran, disciplinas que forman parte (ciencia de la computación, ingeniería de software, sistemas de información, ingeniería en computación) o aportan a la misma el problema de la complejidad.</p>

### **3. Programación I**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

Interpretar críticamente las especificaciones recibidas.

Redactar código.

Reutilizar elementos ya existentes.

Testear el código

Optimizar el código.

<b>Capacidades a desarrollar</b>	<b>Contenidos</b>
<ul style="list-style-type: none"><li>● Aplicar métodos inductivos, deductivos y recursivos en la resolución de situaciones problemáticas y demostraciones matemáticas.</li><li>● Utilizar el pensamiento lógico para resolver un problema de diferentes maneras.</li><li>● Conocer la terminología relacionada con los algoritmos; así como la importancia de aplicar técnicas y estrategias adecuadas de resolución aplicadas a la programación.</li><li>● Aplicar diferentes estructuras de datos en la construcción de algoritmos para la resolución de situaciones problemáticas.</li><li>● Tener capacidad de razonamiento para resolver problemas simples de modo de poder identificar datos, procesos posibles y resultados a obtener.</li><li>● Aplicar razonamiento lógico en la resolución de problemas de estructura condicional, repetitiva de modo de llegar a plantear las distintas alternativas en que puede dividirse un algoritmo.</li><li>● Usar en forma correcta las instrucciones condicionales provistas por un lenguaje de programación de propósito general, para poder plantear programas cuya lógica abarque diversas alternativas y situaciones.</li><li>● Aplicar herramientas para trabajo en equipo lo que permite a los desarrolladores colaborar y realizar cambios en proyectos compartidos y a llevar un seguimiento detallado del progreso.</li></ul>	<p>Concepto de algoritmo, resolución algorítmica de problemas, estrategias de diseño, de implementación, de depuración. Algoritmos fundamentales, algoritmos numéricos simples. Estructuras fundamentales, variables, tipos, expresiones y asignaciones, entrada/salida, estructuras de control condicionales iterativas, funciones y pasaje de parámetros, descomposición estructurada. Concepto de lenguaje de alto nivel y la necesidad de traducción, comparación entre compiladores e intérpretes, aspectos de la traducción dependientes y no dependientes de la máquina. Programas generadores de código. Representación de datos numéricos, rango, precisión y errores de redondeo. Arreglos. Representación de datos de caracteres, listas y su procesamiento. Recolección de espacios no utilizados. La elección de una estructura de datos adecuada. Elementos de complejidad de algoritmos. Declaraciones y tipos, la concepción de tipos como conjunto de valores junto con operaciones, modelo de declaración, elementos de verificación de tipos, tipos y polimorfismo paramétrico. Introducción a la Programación Orientada a Objeto. Entorno de Desarrollo Integrado (IDE). Introducción a C#. Obtención e instalación de un IDE profesional Visual Studio. Desarrollo de software en equipo mediante repositorio Git-Hub. Que es C#, antecedentes y su evolución. Primer programa en lenguaje C# (Hola Mundo).</p>

	<p>Entrada y Salida. Grabación y lectura de un archivo tipo xml.</p> <p>Software y hardware libre y propietario</p>
--	---

#### **4. Base de Datos**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Utilizar ambientes de desarrollo.

<b>Capacidades a desarrollar</b>	<b>Contenidos</b>
<ul style="list-style-type: none"> <li>● Valorar las características y ventajas de las bases de datos, teniendo en cuenta las implementaciones actuales.</li> <li>● Comprender el proceso interno del DBMS en función del pedido del usuario.</li> <li>● Aplicar técnicas de almacenamiento y recuperación de datos.</li> <li>● Distinguir los aspectos de estructura, integridad y manipulación de datos para favorecer su manipulación en el contexto de restricciones del modelo.</li> <li>● Resolver problemas con el diseño de bases de datos relacionales en distintas situaciones problemáticas que necesitan estructuras de datos normalizadas.</li> <li>● Interpretar estructuras de datos ya definidas para lograr analizarlas y mejorarlas, pensando en un equipo de trabajo en los ambientes de desarrollo.</li> <li>● Utilizar la potencialidad del lenguaje SQL como lenguaje para la definición y manipulación de datos en el modelo</li> </ul>	<p>Concepto de almacenamiento y recuperación de información, captura y representación, aplicaciones, búsqueda, recuperación, vinculación, navegación. Metadatos o esquemas asociados con los datos objeto del procesamiento. Problemas de escalabilidad, eficiencia y efectividad.</p> <p>Tipos de Bases de Datos. Estructura de la información, datos y metadatos, lógica y física. Estructura de una Base de Datos. Sistema de Base de Datos, componentes. Administración de Datos y Administración de Bases de Datos. Responsabilidades del DBA y funciones del DBMS. Arquitectura de los sistemas de base de datos. Privacidad, integridad, seguridad y preservación de la información. La persistencia e integridad de los datos.</p> <p>Modelización de datos, modelos conceptuales (E/R, UML), modelo orientado a objetos, modelo relacional, modelos semiestructurados (XML). Concepto y evolución de los sistemas de bases de datos, sus componentes, funciones de un sistema de base de datos.</p>

<p>relacional en un ambiente de desarrollo.</p>	<p>Modelo Relacional. Nociones básicas Dominio, grado y cardinalidad. Propiedades de la relación. Tipos. Claves. Integridad Referencial. Relación entre claves principales / claves foráneas.</p> <p>Análisis y relación entre modelo lógico y conceptual de la Base de Datos.</p> <p>Lenguajes de consulta (SQL, QBE), definición de datos, álgebra relacional, formulación de consultas, lenguaje de actualización, restricciones, integridad. SQL embebido en un lenguaje imperativo. "Script", introducción a un lenguaje de consulta de objetos. Procedimiento almacenados.</p> <p>Diseño de base de datos, dependencia funcional, formas normales, descomposición de un esquema, claves primarias y secundarias. Procesamiento de transacciones, fallas y recuperación, control de recurrencia. Bases de datos distribuidas, problemas que surgen de su explotación.</p>
---	--

## **1ER. CUATRIMESTRE**

### **5. Competencias Comunicacionales I**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

Aprender nuevos lenguajes y tecnologías.

Apropiarse de su actualización personal en las tendencias y demandas del mercado

Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto:

esto comprende:

Escuchar y reformular conceptos en interrelación dialogada.

Elaborar interrogantes coherentes al problema o proyecto planteado

Recibir indicaciones

Jerarquizar metas y determinar secuencias y plazos de realización

Planificar sus actividades.  
 Controlar actividades

- Documentar en base a normas o estándares establecidos

esto comprende:

Registrar actividades realizadas.

Documentar todos los productos de su labor con los requerimientos funcionales y técnicos y de normas de calidad.

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>• Comprender la comunicación humana desde una concepción amplia, integral y compleja.</li> <li>• Reconocer, distinguir y aplicar las competencias comunicacionales orales y escritas.</li> <li>• Demostrar habilidades tanto en la comprensión y producción de textos como en oratoria, de acuerdo al contexto y destinatario.</li> <li>• Diferenciar las particularidades, los roles, la dinámica y los objetivos de los grupos y de los equipos.</li> <li>• Identificar y poner en práctica los aspectos fundamentales y específicos de los equipos de proyectos de desarrollo de software.</li> </ul>	<p>La comunicación, incluyendo la comprensión y producción de textos; las técnicas de indagación destinadas tanto a la búsqueda de información y conocimientos como a interpretar requerimientos de terceros. Medios, ruidos que afectan a la comunicación. Importancia de la retroalimentación. Oratoria. Pitch. Habilidades comunicacionales orales de acuerdo al contexto y destinatario.</p> <p>Conceptos de dinámicas de grupos, grupo y equipos de trabajo, características distintivas. La tarea como eje de la convocatoria de todo grupo/equipo. Tarea explícita e implícita. Dinámica de lo grupal. La mutua representación interna, espacio y tiempo. Objetivos grupales y metas individuales. Lo individual versus lo grupal. Roles y estereotipos, rotación de roles. La empatía. La escucha activa. Conflictos, técnicas de resolución alternativa. El equipo de proyectos de desarrollo de software, roles y responsabilidades de sus integrantes.</p>

## **6. Aproximación al mundo del trabajo**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

Aprender nuevos lenguajes y tecnologías.

Apropiarse de su actualización personal en las tendencias y demandas del mercado

Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Conocer los diferentes contextos de trabajo.</li> <li>● Identificar diferentes procesos de trabajo, sus características, variables puestas en juego, en ambientes reales de trabajo y/o simulados.</li> <li>● Reconocer los diferentes actores que conforman el campo profesional</li> </ul>	<p>Puntos claves: Organización del tiempo, Profesionalidad y marketing personal. Modelo MR3: modelo Realidad Los Dominios de Realidad del ciudadano y profesional. Mundo V.I.C.A. (Volatilidad, Incierto, Complejo y Ambiguo) y respuesta V.I.C.A. (visión, introspección, conocimiento y adaptación) Conducta y prejuicio: principios y conceptos cómo evitar el mundo V.I.C.A.</p> <p>Contrato de trabajo. Características del Contrato individual de trabajo. Derechos y Deberes del trabajador y del empleador. Causas de Extinción del Contrato de trabajo. Convenios Colectivos de Trabajo. Trabajo Freelance: concepto y características.</p> <p>Monotributo: Características de este Régimen Simplificado. Ventajas y Desventajas del Monotributo. Consejo Profesional de Ciencias Informáticas de la Provincia de Córdoba. Registro de Profesionales. Beneficios. Oportunidades Laborales. Capacitaciones</p> <p>Derechos de Autor: normas y principios que la regulan. Copyright y Copyleft: Conceptos y campo de aplicación. Software Libre: Concepto y desarrollo. Sistema Operativo: GNU/Linux: concepto y desarrollo. Creative Commons: Concepto. Objetivos e Influencias. Licencias Creative Commons: Condiciones de uso y combinaciones derivadas de este tipo de licencias.</p> <p>Concepto de Microemprendimiento. Microemprendimientos en Argentina. Factores de éxito y de fracaso de un micro emprendedor. Personalidad de un Micro emprendedor. Alternativas para empezar un negocio. Segmentación de Mercado. Formas de Segmentación. La futura competencia. Producto: concepto. Ciclo de</p>



	<p>Vida de un producto. Valor de un producto. Formas de determinación del Valor - Precio. Logística de distribución. Impulsión de la Organización y del Producto. Indicadores económicos - financieros. Tabla de acciones e indicadores para la Gestión Ética. Responsabilidad Social empresarial. Responsabilidad con el Medio Ambiente. Contexto normativo: responsabilidades empresarias, contratos, responsabilidades del trabajador, leyes de protección de datos personales, propiedad intelectual del software y de contenidos, concepto jurídico aplicable a delitos informáticos. Sindicatos. El convenio colectivo como ámbito de las relaciones laborales. Concepto de paritarias. Colegios profesionales.</p>
--	---

## **2DO. CUATRIMESTRE**

### **7. Arquitectura de las Computadoras**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Reutilizar elementos ya existentes.
- Redactar código.
- Testear el código
- Optimizar el código.

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

- Aprender nuevos lenguajes y tecnologías.
- Apropiarse de su actualización personal en las tendencias y demandas del mercado

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Relacionar las memorias a través de los canales de comunicación de direcciones y datos con el propósito de transferir datos entre los bloques de memoria y la CPU.</li> <li>● Describir los sistemas de numeración en distintas bases y las operaciones aritméticas digitales para desarrollar los distintos códigos de representación utilizando el sistema de numeración binario.</li> <li>● Aplicar los circuitos lógicos básicos con la finalidad de diseñar circuitos combinatoriales y secuenciales considerando sus distintas tecnologías.</li> <li>● Utilizar los circuitos secuenciales para diseñar los registros de desplazamiento como elementos básicos de memorias de más de un bit.</li> <li>● Vincular los distintos dispositivos que componen la arquitectura de una computadora para analizar arquitecturas alternativas teniendo en cuenta las distintas configuraciones de conexión de las memorias.</li> <li>● Analizar y diferenciar las arquitecturas CISC y RISC para aplicar la conveniencia tecnológica de una y otra de acuerdo a la finalidad de su utilización.</li> <li>● Vincular las distintas partes de una instrucción a los fines de obtener los tipos de modos de direccionamiento de un microprocesador con la finalidad de optimizar la programación en bajo nivel.</li> <li>● Realizar programas en lenguaje de bajo nivel con la finalidad de utilizar los recursos directos del hardware de una computadora, considerando ventajas y desventajas en relación al lenguaje de alto nivel.</li> <li>● Identificar los diferentes dispositivos de entrada/salida asociados a las</li> </ul>	<p>Evolución del computador, su organización y unidades funcionales que lo componen. Arquitectura interna de computadores, unidad central de procesamiento, instrucciones y flujo de la información. Memoria: tipos y niveles de organización de la memoria interna y externa Periféricos, clasificación y utilización. Funcionamiento del programa a nivel de la máquina (principalmente como medio de comprender características de su funcionamiento).</p> <p>Sistemas de archivo (datos, metadatos, operaciones, organización, "buffering", secuenciales y no secuenciales). Índices: contenido y estructura. Técnicas estándares de implementación. Archivos de mapeo de memoria. Sistemas de archivos para propósitos especiales. Denominación, búsqueda, acceso, backups.</p> <p>Introducción a la lógica digital, compuertas lógicas, flip-flops, circuitos. Expresiones lógicas y funciones booleanas. Representación de datos numéricos, aritmética con y sin signo, concepto de overflow. Rango, precisión y errores en aritmética de punto flotante. Representación de caracteres, audio e imágenes. Comprensión de datos.</p> <p>Sistemas de Numeración. Concepto. SN Binario: su conveniencia tecnológica. Sistemas de Numeración Posicionales: binario, decimal, octal, hexadecimal. Conversiones. Aritmética Digital (suma y resta). Complemento a la base: concepto de módulo. Resta usando complementos. Aritmética de punto flotante: normalización IEEE-754 Simple Precisión. Codificación: BCD, ASCII. Ejercicios de resolución de problemas típicos y de aplicación cotidiana. Álgebra Digital: Funciones lógicas. Conceptos de álgebra de los circuitos digitales. Operaciones fundamentales. Funciones y formas canónicas. Álgebra de Boole: Propiedades. Compuertas lógicas</p>

<p>computadoras actuales para aplicar a distintas situaciones de configuración organizacional considerando sus ventajas tecnológicas, limitaciones y aplicabilidad a cada caso.</p>	<p>AND, OR, NAND, NOR y OR-EXCLUSIVA. Su aplicación como parte de las memorias y de otras partes del computador.</p> <p>Circuitos Lógicos Combinacionales y Secuenciales. Circuitos lógicos combinacionales básicos: codificadores, decodificadores, multiplexores, demultiplexores, sumadores, decodificador a display de 7 segmentos. Circuitos secuenciales: Flip-flops. Contadores, Registros de desplazamiento. Simulador LOGISIM para implementación y prueba de todos los circuitos.</p> <p>Tecnología. Memoria del sistema: estáticas, dinámicas, RAM, ROM, Caché. Memorias auxiliares: discos duros y sólidos, otras memorias. Dispositivos de entrada y salida: componentes y funcionamiento. Monitores. Impresoras: no matriciales, matriciales, térmicas, láser, trazadores gráficos. Dispositivos de captura directa de datos: escaners, detectores ópticos, etc. Otros dispositivos tecnológicos modernos.</p> <p>El Microprocesador: Arquitectura de microprocesadores: partes que la componen y funcionamiento (UCP, UC, UAL, BUS, Stack, Instrucciones). Concepto de programa almacenado. El Assembler del micro. Introducción a las plataformas CISC y RISC. Conceptos fundamentales. Conjunto de instrucciones para las arquitecturas típicas Familia INTEL: programación básica en Assembler.</p>
---	---

## **8. Competencias Comunicacionales II**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

Aprender nuevos lenguajes y tecnologías.

Apropiarse de su actualización personal en las tendencias y demandas del mercado

Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto:

esto comprende:

Escuchar y reformular conceptos en interrelación dialogada.  
 Elaborar interrogantes coherentes al problema o proyecto planteado  
 Recibir indicaciones  
 Jerarquizar metas y determinar secuencias y plazos de realización  
 Planificar sus actividades.  
 Controlar actividades

- Documentar en base a normas o estándares establecidos

esto comprende:

Registrar actividades realizadas.  
 Documentar todos los productos de su labor con los requerimientos funcionales y técnicos y de normas de calidad.

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Reconocer la relación e influencia mutua entre la comunicación y las nuevas tecnologías, tanto actuales como históricas.</li> <li>● Conceptualizar las Nuevas Tecnologías de la Información y la Comunicación.</li> <li>● Demostrar una visión crítica y fundamentada acerca del uso, consumo y producción de nuevas tecnologías.</li> <li>● Definir la Comunicación Organizacional y reconocer los elementos centrales que la componen, incluyendo la relación con las nuevas tecnologías.</li> <li>● Relacionar la Comunicación Organizacional con la Cultura, el Aprendizaje y el Clima organizacionales, reconociendo esta vinculación en la práctica.</li> <li>● Identificar la incidencia y el rol de la comunicación en equipos de trabajo dentro de una organización, en particular en posibles contextos</li> </ul>	<p>La comunicación y las nuevas tecnologías: Recorrido histórico y actualidad. Las Nuevas Tecnologías de la Información y la Comunicación: definición, características, surgimiento y desarrollo. Influencia de las NTIC en lo individual y en lo colectivo. Debates, proyecciones, certezas e incertidumbres sobre la comunicación del futuro. Narrativas hipermedia y transmedia. Identidad digital profesional. Redes profesionales. Networking.</p> <p>Comunicación Organizacional: Definición, características e importancia. Comunicación Organizacional interna y externa, formal e informal. La Comunicación Organizacional y las nuevas tecnologías. La comunicación en equipos de trabajo dentro de una organización: tareas, liderazgo, roles, objetivos, conflictos, vinculaciones. Cultura, aprendizaje y clima organizacional.</p> <p>La participación inteligente. Su finalidad e impacto en nuestro medio. Modelos y funciones: un modelo de tres funciones</p>

laborales y profesionales de un desarrollador de software	complementarias Conocer + Crear + Comunicar. Participación inteligente y realidad. Sus relaciones y significados. Beneficios. Aportes desde lo tecnológico/deontológico a una mejora de las relaciones humanas. La profesión y el profesionalismo. Atributos de un profesional que actúa con profesionalismo.
---	---

## **9. Ética y Deontología Profesional**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

Aprender nuevos lenguajes y tecnologías.

Apropiarse de su actualización personal en las tendencias y demandas del mercado

Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto:

esto comprende:

Escuchar y reformular conceptos en interrelación dialogada.

Elaborar interrogantes coherentes al problema o proyecto planteado

Recibir indicaciones

Jerarquizar metas y determinar secuencias y plazos de realización

Planificar sus actividades.

Controlar actividades

<b>Capacidades a desarrollar</b>	<b>Contenidos</b>
<ul style="list-style-type: none"> <li>● Identificar las problemáticas que se vinculan con el valor de los datos en el campo profesional y su incidencia en las posibles actividades que realiza el TS.</li> <li>● Reflexionar y analizar el impacto de las TIC en el desarrollo social y económico del sector productivo.</li> <li>● Conocer las características de las relaciones laborales del sector</li> </ul>	<p>Importancia social y económica de los servicios de tecnología de la información, significado de Internet, valor de la información almacenada para las organizaciones, seguridad. Valor de la información para los individuos, normativa relativa a privacidad y "hábeas data". Bases de datos públicas y privadas. Propiedad de datos empresarios. Secretos comerciales e industriales.</p>

<p>profesional que impactan en su ámbito específico de trabajo</p> <ul style="list-style-type: none"> <li>● Reconocer al profesional en desarrollo de software o soluciones tecnológicas como sujeto obligado profesional y éticamente a denunciar mala praxis.</li> <li>● Concientizar al profesional en el campo de la tecnología experimental, su desarrollo en el dominio de la Información y la construcción de conocimiento en código éticos integrados a la globalización.</li> <li>● Concientizar el uso del conocimiento y la habilidad para consumir contenido digital, en confianza y credibilidad responsable de ética y, el cumplimiento que ejerzamos desde la concepción del proyecto, hasta su implementación y monitoreo</li> </ul>	<p>Concepto de Ética. Ética como ciencia Normativa, Especulativa y como Práctica. Diferencia entre Ética y Moral. Acto Moral: elementos constitutivos. Impedimentos del acto moral. Fuentes de la Moralidad: Objeto, Circunstancias y Fin. La responsabilidad. Tipos de responsabilidad. Mérito y Demérito. La sanción y Tipos de Sanciones. Valores: Concepto. Características de los Valores. Tipos de valores.</p> <p>Internet: concepto y origen. Internet y su evolución. Impacto social de Internet. Bases de datos. Gestión de Base de Datos. Habeas Data: concepto. Protección de Datos Personales en la Argentina. Constitución Argentina: Art. 19 y 43. Ley n° 25326. La Ética aplicada a las Ciencias Informáticas y al Internet.</p> <p>Privacidad de datos personales. Normas que rigen el correo electrónico. Protección legal de la propiedad intelectual (incluyendo software), derecho de reproducción y derechos sobre marcas y patentes. Licencias de fabricación, de uso, GNU y “creative commons”.</p>
--	---

## **SEGUNDO AÑO**

### **10. Inglés**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

Aprender nuevos lenguajes y tecnologías.

Apropiarse de su actualización personal en las tendencias y demandas del mercado

Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto:

esto comprende:

Escuchar y reformular conceptos en interrelación dialogada.  
 Elaborar interrogantes coherentes al problema o proyecto planteado  
 Recibir indicaciones

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Reconocer las estructuras básicas de la lengua: gramática, vocabulario y pronunciación.</li> <li>● Comprender y extraer la información global y específica de textos técnicos relacionados con el mundo de las tecnologías de la información.</li> <li>● Comprender y escribir comandos e instrucciones básicas relacionadas con su ámbito personal y profesional.</li> <li>● Comprender y tomar parte en conversaciones sobre asuntos cotidianos personales y profesionales.</li> <li>● Interpretar textos técnicos y relacionados con su actividad profesional</li> <li>● Dominar programas básicos de traducción asistida por computadora</li> </ul>	<p>Inglés Técnico. Lectura e interpretación de textos e información técnica en inglés. Comprensión y producción de textos de complejidad creciente en inglés para comunicarse solicitando o aportando información técnica por e-mail o en foros y listas de discusión. Vocabulario empresarial. Conceptos específicos de marketing.</p> <p>Elementos no verbales que acompañan al texto. Elementos verbales. Funciones comunicativas del discurso científico-técnico. Reconocimiento de instrucciones y recomendaciones. Comprensión de diálogos sobre ideas principales y secundarias.</p> <p>La construcción nominal. Traducción de frases nominales extraídas de textos técnicos. La construcción verbal. Diferencia en uso y estructura de los tiempos. Verbos modales (can, could, must, should, might, must). El modo Imperativo. La Voz Pasiva.</p> <p>Habilidades de comunicación: entrevista laboral, descripción de su perfil laboral y planteo de un problema que se le ha pedido analizar y resolver.</p>

## **11. Estadística y probabilidad aplicadas**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Diseñar la solución.
- Representar el diseño.
- Aplicar patrones de diseño si corresponde.
- Redactar código.
- Testear el código

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Conocer, aplicar y analizar colecciones de datos mediante herramientas estadísticas</li> <li>● Construir modelos matemáticos para resolver situaciones problemáticas que involucran múltiples datos</li> <li>● Conocer y utilizar herramientas de cálculo probabilístico aplicables a la modelización y predicción de comportamientos inciertos</li> </ul>	<p>Permutaciones y combinaciones, propiedades de los números combinatorios. El teorema binomial. Resolución de relaciones de equivalencia, el teorema maestro. Grafos. Conceptos básicos, recorridos, coloreado de vértices, Árboles y bosques. Grafos dirigidos y redes. Aplicaciones de árboles y grafos (algoritmos de recorrida, organización de índices, topología de redes, cálculo del camino crítico). Matrices y vectores como representación de cambios de estado. Espacio finito de probabilidades, medida de probabilidad, eventos. Probabilidad condicional, independencia, teorema de Bayes. Esperanza matemática, variables aleatorias enteras. Ley de los grandes números. Aplicaciones del cálculo diferencial.</p> <p>Estadística descriptiva. Medidas de posición y de dispersión. Efectos del descarte de datos extremos en las diferentes medidas de posición y dispersión. Distribuciones discretas y continuas. Acumulación por rangos. Recolección de datos para análisis estadísticos, clasificación e interpretación. Series temporales. Matrices Matriz. Suma; propiedades. Resta. Multiplicación de una matriz por un escalar; propiedades. Producto; propiedades. Matriz Inversa; propiedades. Matriz Adjunta.</p> <p>Estudio y análisis de funciones; lineal, cuadrática, exponencial y logarítmicas, gráfica de las mismas. Ecuaciones: conceptos de variables, constantes, ecuaciones lineales, cuadráticas exponenciales y logarítmicas. Álgebra matricial, programación de matrices en algún lenguaje de programación. Funciones recursivas, inducción matemática.</p>



## **12. Modelado y Arquitectura de Software**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Aplicar patrones de diseño si corresponde.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Reutilizar elementos ya existentes.
- Utilizar ambientes de desarrollo.
- Valorar opciones de capacitación continua, tanto propias como del equipo de trabajo.

- Verificar la calidad de las aplicaciones, diseñando casos de prueba unitarios y de integración

esto comprende:

- Considerar las necesidades de cobertura de la prueba.
- Diseñar los casos de prueba.
- Preparar el entorno de pruebas.
- Realizar pruebas unitarias.
- Registrar casos de prueba, datos y resultados de pruebas y acciones correctivas.

- Documentar en base a normas o estándares establecidos

esto comprende:

- Documentar todos los productos de su labor con los requerimientos funcionales y técnicos y de normas de calidad.

<b>Capacidades a desarrollar</b>	<b>Contenidos</b>
<ul style="list-style-type: none"><li>● Distinguir y elegir modelos de proceso de desarrollo de software en función de las fortalezas de cada uno y las diferencias en entornos ágiles adaptativos.</li><li>● Identificar requerimientos de un sistema de información en el marco de las organizaciones como parte fundamental del proceso de desarrollo de software.</li></ul>	<p>Fenómenos actuales del escenario del desarrollo del modelado de sistemas como por ejemplo Big data, IoT, aplicaciones de Inteligencia Artificial, Industria 4.0. ¿qué desafíos implica modelar sistemas operativos en la actualidad en el contexto de estos fenómenos?</p> <p>Fundamentos del modelado de datos. Por qué modelar un proyecto de software. Una</p>

<ul style="list-style-type: none"> <li>● Usar Lenguaje de Modelado para la visualización, especificación, construcción y documentación de sistemas orientados a objetos.</li> <li>● Aplicar los fundamentos del paradigma de orientación a objetos para analizar sistemas de información y construir modelos de software considerando lo propuesto por el Proceso Unificado de Desarrollo.</li> <li>● Aplicar la técnica de historias de usuarios para la identificación de requerimientos en el contexto de proyectos de desarrollo ágiles.</li> <li>● Reconocer la existencia de patrones para el diseño de interfaces de usuario, con la finalidad de aplicarlos en la creación de software, tomando en cuenta una percepción centrada en el usuario.</li> <li>● Aplicar un framework de gestión ágil en proyectos de desarrollo de software.</li> <li>● Modelar arquitecturas de software tomando en cuenta los patrones arquitectónicos disponibles con la finalidad de crear arquitecturas de calidad.</li> <li>● Identificar las actividades necesarias para realizar la transferencia de un producto a sus usuarios finales, con la finalidad de considerar los recursos necesarios para su ejecución en el contexto de un proyecto de desarrollo de software.</li> <li>● Reconocer las diferentes formas de transferencia de un producto de software al ambiente de producción, tomando en cuenta el contexto de despliegue específico para cada producto.</li> </ul>	<p>imagen vale más que mil líneas de código. Ventajas de la modelización. Abstracción de situaciones complejas para ser tratadas bajo modelos más simples</p> <p>Análisis de requerimientos de software, el proceso, partes interesadas. Requerimientos funcionales, prioridades y criterios de realización. Análisis orientado a objetos y UML como herramienta de modelado de sistemas. Desarrollo de proyectos a través de UML. Diagramas de clase. Escenarios, historias y casos de uso; diseño centrado en el usuario. Representación del comportamiento: diagramas de secuencia, máquinas de estado, diagramas de actividad. Diagramas de Flujos de Datos. Elementos para su construcción. Técnicas de implementación. Diagrama de casos de uso: elementos constitutivos, integración y complementariedad con otros diagramas. Descripción de casos de uso. Inclusión y extensión de casos.</p> <p>Diseño orientado a objetos. Diseño estructurado. Diseño orientado al reuso de componentes, incorporación de elementos disponibles.</p> <p>Arquitectura del Software: Estructura de datos. Diagrama de Arquitectura de Sistemas. Diagrama de Arquitectura de Sitios Web. Diagrama de Arquitectura de Aplicaciones. Diagrama de Arquitectura de Software. Decisiones de diseños. Aplicaciones monolíticas. Microservicios. Atributos de calidad del software</p> <p>Análisis de requerimientos de software. Manejo de documentación como producto de la modelización. Implementación de versiones de software. Requerimientos funcionales, prioridades y criterios de realización. Arquitectura de software. Estructuras en el diseño de software (Procesos distribuidos, cliente-servidor)</p> <p>Aspectos importantes del diseño. Diseños orientados a objetos. Diseños centrados en el usuario. Diseño estructurado. Diseño</p>
--	---

	<p>orientado al reuso de componentes. Diseño de base de conocimiento.</p> <p>Aplicación e implementación del diseño. Diseño de una salida eficaz. Ajustes y finalización del diseño. Aseguramiento de la calidad mediante la aplicación de la ingeniería de software.</p> <p>Metodologías ágiles. Análisis de requerimientos de software, el proceso, partes interesadas. Requerimientos funcionales, prioridades y criterios de realización. Análisis orientado a objetos y UML. Diagramas de clase. Escenarios, historias y casos de uso; diseño centrado en el usuario. Representación del comportamiento: diagrama de secuencia, máquinas de estado, diagramas de actividad. Redes de Petri. Pre y post condiciones.</p> <p>Modelado, actividades principales, características esenciales de un modelo, buenas prácticas en la actividad del modelado, documentación como producto de la actividad de modelado.</p> <p>Principios generales de diseño: descomposición, desacoplamiento, cohesión, reusabilidad, portabilidad, testeabilidad, flexibilidad, escalabilidad. Patrones de diseño. Arquitectura de software: concepto de vistas, arquitecturas distribuidas, “pipe -and-filter”, “model-view-controller”.</p> <p>Diseño orientado a objetos. Diseño estructurado. Diseño orientado al reuso de componentes, incorporación de elementos disponibles al diseño</p>
--	---

### **13. Programación II**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Diseñar la solución.

Representar el diseño.  
 Verificar el diseño.  
 Reutilizar elementos ya existentes.  
 Redactar código.  
 Testear el código  
 Optimizar el código.  
 Controlar cambios y versiones.  
 Utilizar ambientes de desarrollo.  
 Valorar opciones de capacitación continua, tanto propias como del equipo de trabajo.

- Optimizar y depurar código de aplicaciones de software para resolver defectos o mejorarlos

esto comprende:

Interpretar código.  
 Diagnosticar defectos.  
 Documentar defectos e Informarlos  
 Depurar defectos.

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

Aprender nuevos lenguajes y tecnologías.  
 Apropiarse de su actualización personal en las tendencias y demandas del mercado  
 Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Aplicar técnicas de programación orientada a objetos, desarrollando sistemas mediante el lenguaje adecuado a tal fin.</li> <li>● 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.</li> <li>● Verificar la funcionalidad de las clases diseñando sus respectivas clases de</li> </ul>	<p>Programación orientada a objetos, encapsulamiento y ocultamiento de información, separación entre comportamiento e implementación, clases y subclases, herencia (sustitución), polimorfismo (subtipos vs. herencia), jerarquía de clases, clases colección y protocolos de interacción.          Tecnología ADO. Componentes, objetos Connection, Command, DataReader, DataAdapter, Conjunto de Datos (DataSet)..          Trabajo Modo Conectado, Modo Desconectado. Ventajas. Desventajas.</p>

<p>prueba, elaborando la documentación técnica correspondiente.</p> <ul style="list-style-type: none"> <li>● Identificar los problemas que se presenten en el desarrollo o programación a partir del análisis, jerarquización y priorización de la información.</li> <li>● 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.</li> <li>● 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.</li> </ul>	<p>Aplicaciones Windows Forms con acceso a Base de Datos, trabajando en capas.</p> <p>Validación de datos. Tipos de Validación (aplicación en proyecto)</p> <p>Máquinas virtuales, concepto, jerarquía de máquinas virtuales, lenguajes intermedios, asuntos de seguridad que surgen al ejecutar código de una máquina diferente.</p> <p>Verificación unitaria de unidades de código, concepto de cubrimiento, organización, ejecución y documentación de la prueba.</p> <p>Manejo de memoria en tiempo de ejecución, punteros y referencias, estructuras encadenadas, pilas, colas y tablas de hashing.</p> <p>Representación de grafos. Algoritmos de camino mínimo, concepto de autómata.</p> <p>Recursión, concepto, funciones matemáticas recursivas, funciones recursivas simples, estrategias de dividir y conquistar, backtracking recursivo.</p> <p>Algoritmo de búsqueda sucesiva y binaria, de ordenamiento con tiempos cuadráticos (selección, inserción), con tiempos <math>O(N \log N)</math> (QuickStart, heapsort, megasort).</p> <p>Tablas de hashing, estrategias para evitar colisiones. Árboles de búsqueda binaria.</p> <p>Estándares de nomenclatura y formato en programación, encabezado de módulos u objetos con comentarios que expliciten en sus alcances y limitaciones, inserción de comentarios o advertencias en el código, documentación adicional.</p> <p>Programación conducida por eventos, métodos por manejo de eventos, propagación de eventos, manejo de excepciones.</p> <p>Programación defensiva, importancia de verificar para evitar el overflow de arreglos y listas. Alternativas o dispositivos de lenguaje de programación para evitarlo. Cómo atacantes pueden utilizar el overflow para destruir el stack en tiempo de ejecución.</p> <p>Concepto de computación paralela, concurrente, multinúcleos.</p>
---	---

## **14. Práctica Profesionalizante I**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Aplicar patrones de diseño si corresponde.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Reutilizar elementos ya existentes.
- Redactar código.
- Testear el código
- Utilizar ambientes de desarrollo.

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto

esto comprende:

- Obtener métricas a partir de los registros de actividades.
- Reportar avances y dificultades.
- Planificar sus actividades.
- Controlar sus actividades.
- Coordinar actividades de acuerdo a roles y funciones del equipo de trabajo.
- Elaborar interrogantes coherentes al problema o proyecto planteado
- Escuchar y reformular conceptos en interrelación dialogada.
- Recibir indicaciones
- Reconocer sus potencialidades personales y ajenas al momento de integrar equipos de trabajo
- Jerarquizar metas y determinar secuencias y plazos de realización

- Documentar en base a normas o estándares establecidos

esto comprende:

- Registrar actividades realizadas.
- Documentar todos los productos de su labor con los requerimientos funcionales y técnicos y de normas de calidad.

El mundo del trabajo, las relaciones que se generan dentro de él, sus formas de organización y funcionamiento y la interacción en las actividades productivas en contextos socioeconómicos locales y regionales, conjugan relaciones tanto socioculturales como económico productivas que sólo pueden ser aprehendidas a través de una participación activa de los estudiantes en un proceso de producción de bienes o servicios.

La adquisición de capacidades para desempeñarse en situaciones sociolaborales concretas sólo es posible si se generan en los procesos educativos actividades formativas de acción y reflexión sobre situaciones reales de trabajo.

En este sentido, el campo de formación de la práctica profesionalizante está destinado a posibilitar la integración y contrastación de los saberes construidos en la formación de los otros campos y garantizar la articulación teoría-práctica en los procesos formativos a través del acercamiento de los estudiantes a situaciones reales de trabajo, propiciando una aproximación progresiva al campo ocupacional hacia el cual se orienta la información. De esta manera los estudiantes toman contacto con diferentes situaciones y problemáticas que permitan tanto la identificación del objeto de la práctica profesional como la del conjunto de procesos técnicos, tecnológicos, científicos, culturales, sociales y jurídicos que se involucran en la diversidad de situaciones socioculturales y productivas que se relacionan con un posible desempeño profesional.

En el marco de un ambiente de trabajo real o simulado las actividades formativas grupales e individuales tienen que integrar prácticas como la interpretación crítica de especificaciones de artefactos de software, el diseño de la solución, su justificación y validación; la construcción de partes no triviales, revisión, verificación unitaria y depuración, aplicando buenas prácticas de programación y documentación; así como también su integración con otros artefactos ya existentes o desarrollados por otros para conformar versiones, incluyendo la depuración de los errores encontrados.

Esto se puede lograr en el sector productivo, realizando acuerdos en los que se planifique y verifique que el estudiante realice un conjunto de tareas del tipo de las descritas, o en la institución educativa, creando ámbitos de desarrollo de software típicamente denominados software Factory, que reproduzca en las condiciones en las que se desarrollan proyectos las empresas del sector, organizando equipos de desarrollo y contando con figuras docentes que asuman el rol de gerentes de desarrollo o responsables por la calidad. También resulta importante contar con un cliente creíble que plantea demandas realistas y que se preste al juego de modificar alguno de los requerimientos durante el proceso.

Esta actividad formativa debe ser cumplida por todos los estudiantes, con supervisión docente, y la institución educativa debe garantizarla durante y a lo largo de la trayectoria formativa.

Se pretende destacar la necesidad de las aptitudes de auto-aprendizaje e investigación de las soluciones más convenientes para cumplir con el cometido y la elaboración de un proyecto de un software aplicando las técnicas aprendidas en otras asignaturas y comparando con las estrategias implementadas por iniciativa propia durante el año lectivo.

Para lograrlo, el estudiante, elaborará un proyecto a elección y de complejidad evolutiva, con consumo de datos contextualizado en una realidad-problema desafiante. Este proyecto deberá implicar: recolección de requerimientos, desarrollo de front-end, confección y creación de bases de datos, consumo de datos desde el back-end, consumo de APIs, siempre apoyado en el trabajo en equipo y la comunicación fluida con los integrantes del grupo, manejando herramientas que permitan esta organización, y la utilización de diferentes marcos de referencias para el trabajo organizado. Se establecerán entregas parciales en periodos de tiempos establecidos según el marco de referencia utilizado (scrum, kanban o el elegido para el periodo lectivo), permitiendo el acercamiento de los estudiantes a situaciones reales de trabajo que requieren de habilidades duras y el desarrollo de habilidades blandas.

El docente estará en un rol de guía en el transcurso de las diferentes etapas establecidas y un rol de cliente al momento de la finalización del periodo y defensa de los avances (cierre del periodo establecido).

El o los proyectos que se desarrollen formarán parte, no excluyente de otras posibilidades, del portfolio de iniciativas a madurar en la Práctica Profesionalizante II.

## 1ER. CUATRIMESTRE

### 15. Sistemas Operativos

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Reutilizar elementos ya existentes.
- Redactar código.
- Testear el código
- Optimizar el código.

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

- Aprender nuevos lenguajes y tecnologías.
- Apropiarse de su actualización personal en las tendencias y demandas del mercado

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"><li>● Reconocer las principales arquitecturas y servicios de los sistemas operativos para utilizarlos correctamente.</li><li>● Comparar los algoritmos de planificación de procesos con el fin de ejemplificar su funcionamiento en un sistema operativo en particular</li><li>● Aplicar los comandos de administración para experimentar su uso, tomando en cuenta el sistema operativo Windows y Linux.</li><li>● Aplicar la virtualización de las direcciones de memoria, con la finalidad de distinguir los métodos de gestión de la memoria, de acuerdo a las tecnologías disponibles de los microprocesadores.</li></ul>	<p>Los sistemas operativos, su papel y propósito, funcionalidades típicas. Mecanismo que soportan los modelos cliente servidor y otros dispositivos. Características y objetivos de su diseño y su influencia en la seguridad, interoperabilidad, capacidad multimedial. Estructuras de sistemas operativos (monolíticos, modulares y de "micro-kernel"). Abstracciones, procesos y recursos. Organización de los dispositivos, interrupciones: métodos e implementación. Concepto de Estado usuario/supervisor y protección, transición al modo supervisor. Estados y transiciones; cola de procesos, bloque de control de procesos. Despacho, "switching" de contexto, "switching" cooperativo y "preempted". Ejecución concurrente: ventajas y desventajas. El</p>



<ul style="list-style-type: none"> <li>● Aplicar los comandos de administración de memoria para experimentar su uso, tomando en cuenta el sistema operativo Windows y Linux.</li> <li>● Analizar la importancia de la seguridad de un sistema con la finalidad de mantener la operación del mismo según el diseño considerado óptimo por el proveedor del Sistema Operativo</li> <li>● Identificar los diferentes tipos de software malicioso con el fin de detectar los riesgos a los que está expuesto el sistema en función de la realidad actual</li> <li>● Instalar y configurar máquinas virtuales en diferentes plataformas en función de las necesidades del cliente en particular</li> </ul>	<p>problema de la exclusión mutua y algunas soluciones. Bloqueos: causas, condiciones, prevención.</p> <p>Paso de mensajes sincrónico y asincrónico. Problema de consumidor-productor y sincronización (mutex, semáforos) lector-escritor y “filósofos comensales”: comparación y análisis de condiciones de cada problema.. Multiprocesamiento (interrupción de ciclos, reentrada). Política de despacho de procesos; programación con y sin prioridades de interrupción. Procesos y “threads”. Elementos de tiempo real y tiempo límite.</p> <p>Administración de memoria. Revisión de memoria física y hardware de administración de memoria. Paginamiento y memoria virtual “Working sets” y “trashing”. “Cacheo”. Administración de dispositivos, características de dispositivos seriales y paralelos. Abstracción de diferencia entre dispositivos. Estrategias de “buffering”. Acceso directo a memoria. Recuperación de fallas.</p> <p>Seguridad y protección. Políticas y mecanismos de separación. Métodos y dispositivos de seguridad. Protección, control de acceso y autenticación. Backups. “Scripting”. Comandos básicos del sistema, creación de “scripts”, pasaje de parámetros. Ejecución de un “script”.</p> <p>Ambientes gráficos para edición, editores inteligentes. Herramientas integradas disponibles para la edición en distintos lenguajes y ambientes. Bibliotecas de clases, programas y rutinas.</p> <p>Sistemas de archivo (datos, metadatos, operaciones, organización, “buffering”, secuenciales y no secuenciales). Índices: contenido y estructura. Técnicas estándares de implementación. Archivos de mapeo de memoria. Sistemas de archivo para propósitos especiales. Denominación, búsqueda, acceso, backups.</p> <p>Práctica formativa: Ejercitación en Windows y Linux: Instalación de SO Windows y Linux en máquinas virtuales para realizar</p>
---	---

	<p>prácticas en modo comando o consola. Experimentación, de manera articulada con Programación, la reconstrucción simplificada de algunos algoritmos que se aplican en módulos centrales de los sistemas operativos como el de “Gestor de proceso” que permita a los estudiantes comprender el funcionamiento básico del sistema operativo desde una perspectiva más compleja (y no sólo como usuario o desde un abordaje teórico) que le permita diseñar e incluso programar de qué manera el software lleva adelante algunas de sus funciones básicas.</p>
--	--

## 2DO. CUATRIMESTRE

### 16. Redes

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Reutilizar elementos ya existentes.
- Redactar código.
- Testear el código
- Optimizar el código.

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

- Aprender nuevos lenguajes y tecnologías.
- Apropiarse de su actualización personal en las tendencias y demandas del mercado

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Establecer en qué casos utilizar cada tipo de red según las necesidades de una organización</li> <li>● Conocer la arquitectura TCP/IP con la finalidad de descubrir los protocolos y servicios utilizados actualmente</li> <li>● Identificar los métodos de acceso al medio para conocer la implementación de una red según los estándares actuales de redes locales</li> <li>● Aplicar el direccionamiento IPv4 con el objetivo de diseñar un esquema de direccionamiento según las necesidades de una organización en particular.</li> <li>● Realizar cálculos de direccionamiento IP con el fin de satisfacer los requerimientos solicitados en un caso concreto.</li> <li>● Comparar los protocolos IPv4 e IPv6 con el fin de descubrir sus ventajas y desventajas teniendo en cuenta su implementación en un caso particular.</li> <li>● Contrastar el direccionamiento estático y dinámico para discernir en qué casos aplicarlos según los requerimientos de una determinada organización.</li> <li>● Utilizar herramientas de administración de una red con la finalidad de aplicarlas en la configuración y resolución de problemas básicos de conectividad.</li> <li>● Determinar la importancia de la seguridad para aplicarla en el diseño de una red informática, según las políticas de seguridad de una organización.</li> <li>● Reconocer los elementos claves sobre seguridad con el fin de recomendar su aplicación en el diseño de una red segura.</li> </ul>	<p>Orígenes y evolución de Internet y las comunicaciones digitales. Arquitecturas de Red. Especializaciones en la computación centrada en redes. Redes y protocolos. Computación distribuida. Paradigma cliente/servidor y peer to peer. Computación sin cables y móvil. Estándares de redes y cuerpos de estandarización. Modelos de referencia: modelo OSI, TCP/IP. Espacio de direcciones del protocolo, categorías de direcciones. Máscara de Red. Elementos de ruteo e interconexión. Aspectos de seguridad de redes.</p> <p>Aplicaciones en redes. Protocolos a nivel de la capa de aplicación. Interfaces web: “browsers” y APIs. Subprotocolos TCP y UDP. El “socket” como atracción. Concepto de paquete “Polling” con primitivas bloqueantes. RCP. “Objects brokers”.</p> <p>Aspecto de administración de redes, uso de contraseñas y mecanismos de control de acceso, servidores de nombres de dominios y de servicios, proveedores de servicios en Internet. Aspectos de seguridad y firewalls. Asuntos de calidad de servicio: comportamiento, recuperación de fallos.</p> <p>Concepto de cloudHosting (servidor en la nube), instalación de paquetes de servicios, servido de base de datos, servidor web, lenguajes de programación (php python, javascript), aplicación de framework de trabajo, django, node.js. Prácticas aplicadas con placas de desarrollo con Raspberry Pi.</p>

## TERCER AÑO

### 17. Interfaz del Usuario

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Reutilizar elementos ya existentes.
- Redactar código.
- Testear el código
- Optimizar el código.
- Controlar cambios y versiones.
- Utilizar ambientes de desarrollo.
- Valorar opciones de capacitación continua, tanto propias como del equipo de trabajo.

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

- Aprender nuevos lenguajes y tecnologías.
- Apropiarse de su actualización personal en las tendencias y demandas del mercado
- Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Conocer todos los elementos básicos para la producción de elementos gráficos para las distintas interfaces de usuarios.</li> <li>● Manejar el lenguaje estético gráfico - tecnológico - digital.</li> <li>● Diseñar interfaces que respondan a determinados requerimientos de las aplicaciones y capacidades de los usuarios del sistema.</li> <li>● Utilizar herramientas para crear el diseño y la interacción de aplicaciones.</li> <li>● Desarrollar prototipos y Wireframes que permita compartirlos con el resto del equipo para obtener las especificaciones técnicas para el desarrollo del software</li> </ul>	<p>Interacción ser humano-máquina, conceptos básicos. Distintos contextos para interfaces: visuales o de texto en aplicaciones habituales, interfaces web con dispositivos para navegación, sistemas colaborativos, juegos y otras aplicaciones multimediales, interfaces con o por medio de diversos dispositivos, lo que pueden incluir teléfonos móviles y TV digital.</p> <p>Proceso de desarrollo centrado en el usuario: foco temprano en los usuarios, prueba empírica de la calidad, diseño iterativo. Medidas de evaluación: utilidad, eficiencia, facilidad de aprendizaje, satisfacción del usuario. Modelos de diseño de la interacción: atención, movimiento, cognición, percepción y reconocimiento.</p>

	<p>Diseño para el cambio: adaptación a otras lenguas o localismos, adaptación a la diversidad de condiciones humanas. Notación para especificar interfaces. El manejo de los errores de los usuarios del sistema. Técnicas y herramientas de prototipado.</p> <p>Principios de interfaces gráficas, acciona-objeto vs. objeto-acción, eventos en interfaces de usuario, estándares, errores más comunes. Interfaces para un sistema nativo, uso del browser para sistemas que operan en la web. Utilización de librerías de estilos para diseño de sitios y aplicaciones.</p> <p>Manejo de errores del usuario y del sistema. Técnicas y herramientas de prototipado. La importancia del diseño universal. Tipos de discapacidades y soluciones. Accesibilidad en la web. Tecnología, Diversidad y Accesibilidad. El reto de la Diversidad. Estrategias para afrontar la diversidad. Accesibilidad Web. Evaluación de la Accesibilidad</p> <p>Diseño UX/UI, Patrones de diseño &amp; contenidos, Prototipo funcional y métricas UX.</p> <p>Sistema de grilla, concepto de User Flow.</p>
--	--

## **18. Ingeniería de Software**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Aplicar patrones de diseño si corresponde.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Reutilizar elementos ya existentes.
- Redactar código.
- Testear el código
- Utilizar ambientes de desarrollo.

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto

esto comprende:

- Obtener métricas a partir de los registros de actividades.
- Reportar avances y dificultades.
- Planificar sus actividades.
- Controlar sus actividades.
- Coordinar actividades de acuerdo a roles y funciones del equipo de trabajo.
- Elaborar interrogantes coherentes al problema o proyecto planteado
- Escuchar y reformular conceptos en interrelación dialogada.
- Recibir indicaciones
- Reconocer sus potencialidades personales y ajenas al momento de integrar equipos de trabajo
- Jerarquizar metas y determinar secuencias y plazos de realización

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Distinguir entre los diferentes ciclos de vida para el desarrollo de software, sus ventajas y desventajas en el marco de la Administración de Proyectos de Software.</li> <li>● Analizar comparativamente los enfoques de gestión tradicionales basados en procesos definidos con los enfoques de gestión basados en procesos empíricos con el objetivo de elegir el más adecuado a un contexto particular.</li> <li>● Interpretar la filosofía ágil para poder determinar el enfoque en un contexto particular de desarrollo de software.</li> <li>● Calcular métricas para obtener visibilidad en el contexto de proyectos de desarrollo de software.</li> <li>● Utilizar frameworks Ágiles para gestionar productos de software acordes a las expectativas de los involucrados.</li> <li>● Aplicar la técnica de user stories para la identificación de requerimientos en el contexto de proyectos de desarrollo ágiles</li> <li>● Utilizar los principales modelos de calidad de software existentes en el mercado para poder evaluar cuál es el</li> </ul>	<p>Programas de trabajo y resolución conjunta de problemas. Modelos del ciclo de vida del software y de procesos de desarrollo. El problema del mantenimiento y las migraciones de plataforma.</p> <p>Metodologías tradicionales y ágiles. Metodologías ágiles, concepto de “sprint”, fraccionamiento del producto en unidades realizables en un “sprint”, cola dependientes, mejora de productos provisorios (refactoring) variación de los roles y la documentación en el marco de un proceso en el que se aplican metodologías ágiles.</p> <p>Gestión en los cambios, concepto de versión, “Guild”, producto de la asignación. Concepto de componente. Elementos de administración de la configuración y control de versiones de software. Herramientas de versionado. Otras herramientas (bibliotecas, diccionarios, repositorios) del proyecto.</p> <p>El problema de asegurar la calidad y elementos de métrica. Modelo de madurez de la capacidad de desarrollo. Enfoques para la mejora del proceso, métricas. El proceso personal de software, estadísticas personales y capitalización de experiencias. Gestión de proyectos de software: herramientas. Estimaciones.</p>

<p>más adecuado para aplicar en un contexto particular.</p> <ul style="list-style-type: none"> <li>● Elegir herramientas de versionado de código para compartir el código fuente y mantener un registro de los cambios en el desarrollo del trabajo grupal en el marco básico de controlar los cambios de cada integrante del grupo.</li> <li>● Identificar técnicas y herramientas para hacer aseguramiento de calidad de software en los proyectos de desarrollo de software.</li> <li>● Utilizar técnicas (auditorías, revisión e inspecciones de software) relacionadas con el aseguramiento de la calidad del proceso y del producto con la finalidad de entregar un producto de software de calidad.</li> </ul>	
---	--

### **19. Programación III**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

Interpretar críticamente las especificaciones recibidas.

Interpretar la arquitectura del sistema en el cual se inserta la asignación.

Diseñar la solución.

Representar el diseño.

Verificar el diseño.

Reutilizar elementos ya existentes.

Redactar código.

Testear el código

Optimizar el código.

Controlar cambios y versiones.

Utilizar ambientes de desarrollo.

Valorar opciones de capacitación continua, tanto propias como del equipo de trabajo.

- Optimizar y depurar código de aplicaciones de software para resolver defectos o mejorarlos

esto comprende:

Interpretar código.

Diagnosticar defectos.  
 Documentar defectos e Informarlos  
 Depurar defectos.

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

Aprender nuevos lenguajes y tecnologías.  
 Apropiarse de su actualización personal en las tendencias y demandas del mercado  
 Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

Capacidades a desarrollar	Contenidos
<ul style="list-style-type: none"> <li>● Diseñar y programar aplicaciones web y la integración con otros sistemas o plataformas.</li> <li>● Conocer y aplicar criterios de seguridad y prevención de vulnerabilidades en el desarrollo web</li> <li>● Diseñar y construir interfaces amigables y funcionales en dispositivos móviles.</li> <li>● Desarrollar aplicaciones que incluyan el acceso a recursos multimedia de otros dispositivos</li> <li>● Crear código web reutilizable y bibliotecas para su desarrollo</li> <li>● Procesar la información recibida a través del front end</li> <li>● Comprender el proceso de comunicación entre cliente y servidor web</li> <li>● Diferenciar entre páginas web estáticas de páginas web dinámicas</li> <li>● Comprender la estructura de una página HTML, sus etiquetas, atributos, el diseño de formularios</li> <li>● Crear páginas web estáticas respetando los estándares de la W3C</li> <li>● Incorporar el diseño de interfaz gráfica con imágenes a la/s página/s web generadas</li> <li>● Entender la estructura de un archivo CSS de hojas de estilo en cascada para la modificación de atributos de estilo y la definición de clases</li> </ul>	<p>Diseño de interacciones software-usuario. Diseño pantallas. Diseño de interfaces para la web con ayudas para la navegación. Diseño de interfaces para alguna norma estándar (USB, "bluetooth") para dispositivos.</p> <p>Desarrollo de software en ambientes distribuidos.</p> <p>Tecnologías web. Lenguajes de programación utilizados para el desarrollo de páginas y sistemas web.</p> <p>Principios de ingeniería web. Sitios web estructurados mediante bases de datos.</p> <p>Tecnologías de búsqueda en web. El papel del "middleware", herramienta de apoyo.</p> <p>Aplicaciones basadas en tecnologías para toda la empresa. Aplicaciones cooperativas. Sistemas de "workflow". Herramientas para desarrollo en ambientes web. "Frameworks" de aplicaciones y su utilización. Creación y administración de sitios web.</p> <p>Patrón MVC para desarrollar aplicaciones web. Programación back-end, diseño de modelos, controladores, métodos, validación de contenido, creación de APIs, Migraciones automáticas y manuales. Utilizaciones de objetos con clases tanto para vistas como para almacenamiento de</p>



<ul style="list-style-type: none"> <li>• Escribir programas con JavaScript para ejecución del lado del navegador, que permita utilizar las instrucciones, estructuras, almacenamiento en memoria</li> <li>• Realizar operaciones de lectura y generación de datos en formato JSON</li> </ul>	<p>información en bases de datos. Métodos de consultas sobre bases de datos.</p> <p>Seguridad en aplicaciones, control de usuarios, permisos y niveles de acceso.</p> <p>Diseño front-end utilizando el lenguaje de maquetado HTML, el lenguaje de diseño y estilo CSS para las vistas de la web con ayudas para la navegación , como así también el uso de JavaScript para diferentes funciones y algoritmos del lado del usuario.</p> <p>Herramientas de versiones de código fuente, para poder trabajar en equipo dividiendo las tareas y centralizando el desarrollo del código fuente. Creación de ventanas de diálogo. Depuración de errores. Validación de datos de formularios. Formato JSON para intercambio de datos e instrucciones para su manipulación</p>
--	---

## **20. Práctica Profesionalizante II**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Modelizar y construir aplicaciones de software a partir de especificaciones

esto comprende:

- Interpretar críticamente las especificaciones recibidas.
- Interpretar la arquitectura del sistema en el cual se inserta la asignación.
- Aplicar patrones de diseño si corresponde.
- Diseñar la solución.
- Representar el diseño.
- Verificar el diseño.
- Reutilizar elementos ya existentes.
- Redactar código.
- Testear el código
- Utilizar ambientes de desarrollo.

- Verificar la calidad de las aplicaciones, diseñando casos de prueba unitarios y de integración

esto comprende:

- Considerar las necesidades de cobertura de la prueba.
- Diseñar los casos de prueba.
- Preparar el entorno de pruebas.
- Realizar pruebas unitarias.

Registrar casos de prueba, datos y resultados de pruebas y acciones correctivas.

- Optimizar y depurar código de aplicaciones de software para resolver defectos o mejorarlos

esto comprende:

- Interpretar código.
- Diagnosticar defectos.
- Documentar defectos e Informarlos
- Depurar defectos.

- Documentar en base a normas o estándares establecidos

esto comprende:

- Registrar actividades realizadas.
- Documentar todos los productos de su labor con los requerimientos funcionales y técnicos y de normas de calidad.

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto

esto comprende:

- Obtener métricas a partir de los registros de actividades.
- Reportar avances y dificultades.
- Planificar sus actividades.
- Controlar sus actividades.
- Coordinar actividades de acuerdo a roles y funciones del equipo de trabajo.
- Elaborar interrogantes coherentes al problema o proyecto planteado
- Escuchar y reformular conceptos en interrelación dialogada.
- Recibir indicaciones
- Reconocer sus potencialidades personales y ajenas al momento de integrar equipos de trabajo
- Jerarquizar metas y determinar secuencias y plazos de realización

- Determinar y proyectar su propia actualización profesional, en relación a la transformación del entorno laboral y tecnológico, a partir de la capacitación continua

esto comprende:

- Aprender nuevos lenguajes y tecnologías.
- Apropiarse de su actualización personal en las tendencias y demandas del mercado
- Involucrarse responsablemente en la actualización profesional atento las demandas y requerimientos derivados de la actualización de nuevos productos y tendencias del mercado

Se espera que durante el cursado del espacio curricular los estudiantes sean capaces de:

- Integrar los conocimientos teóricos-prácticos y habilidades adquiridas a lo largo de la carrera, con el fin de aplicarlos en un caso real, que abarque todo el ciclo de desarrollo de los Sistemas de Información.
- Desarrollar habilidades para trabajar en equipo, con el fin de poder incorporarse a equipos en diferentes ámbitos, trabajar de manera interdisciplinaria y colaborativa y capitalizar la sinergia que se produce en ese estilo de trabajo, para alcanzar los objetivos fijados.
- Reconocer las necesidades de información de una organización o contexto real, para realizar su análisis asumiendo posturas críticas, frente a la actividad y procesos existentes.
- Desarrollar un sistema con la finalidad de producir una solución innovadora, que responda a las necesidades de información de una organización o contexto real.
- Desarrollar habilidades de comunicación, mediante la realización de exposiciones sobre el proyecto, a fin de dar a conocer su estado de avance, el sistema, sus características, las tecnologías aplicadas y demostrar el funcionamiento de la solución.

El mundo del trabajo, las relaciones que se generan dentro de él, sus formas de organización y funcionamiento y la interacción en las actividades productivas en contextos socioeconómicos locales y regionales, conjugan relaciones tanto socioculturales como económico productivas que sólo pueden ser aprehendidas a través de una participación activa de los estudiantes en un proceso de producción de bienes o servicios.

La adquisición de capacidades para desempeñarse en situaciones sociolaborales concretas sólo es posible si se generan en los procesos educativos actividades formativas de acción y reflexión sobre situaciones reales de trabajo.

En este sentido, el campo de formación de la práctica profesionalizante está destinado a posibilitar la integración y contrastación de los saberes construidos en la formación de los otros campos y garantizar la articulación teoría-práctica en los procesos formativos a través del acercamiento de los estudiantes a situaciones reales de trabajo, propiciando una aproximación progresiva al campo ocupacional hacia el cual se orienta la información. De esta manera los estudiantes toman contacto con diferentes situaciones y problemáticas que permitan tanto la identificación del objeto de la práctica profesional como la del conjunto de procesos técnicos, tecnológicos, científicos, culturales, sociales y jurídicos que se involucran en la diversidad de situaciones socioculturales y productivas que se relacionan con un posible desempeño profesional.

En el marco de un ambiente de trabajo real o simulado las actividades formativas grupales e individuales tienen que integrar prácticas como la interpretación crítica de especificaciones de artefactos de software, el diseño de la solución, su justificación y validación; la construcción de partes no triviales, revisión, verificación unitaria y depuración, aplicando buenas prácticas de programación y documentación; así como también su integración con otros artefactos ya existentes o desarrollados por otros para conformar versiones, incluyendo la depuración de los errores encontrados.

Esto se puede lograr en el sector productivo, realizando acuerdos en los que se planifique y verifique que el estudiante realice un conjunto de tareas del tipo de las descritas, o en la institución educativa, creando ámbitos de desarrollo de software típicamente denominados software Factory, que reproduzca en las condiciones en las que se desarrollan proyectos las empresas del sector, organizando equipos de desarrollo y contando con figuras docentes que asuman el rol de gerentes de desarrollo o responsables por la calidad. También resulta

importante contar con un cliente creíble que plantea demandas realistas y que se preste al juego de modificar alguno de los requerimientos durante el proceso.

Esta actividad formativa debe ser cumplida por todos los estudiantes, con supervisión docente, y la institución educativa debe garantizarla durante y a lo largo de la trayectoria formativa.

Se pretende destacar la necesidad de las aptitudes de auto-aprendizaje e investigación de las soluciones más convenientes para cumplir con el cometido y la elaboración de un proyecto de un software aplicando las técnicas aprendidas en otras asignaturas y comparando con las estrategias implementadas por iniciativa propia durante el año lectivo.

Se desarrollará una aplicación web y/o mobile completa, desde el análisis, la propuesta, el diseño de interfaz, la programación y el almacenamiento en una base de datos. Se deberán planificar entregas de las diferentes etapas del desarrollo, donde el docente se posiciona como cliente. Si se considera adecuado, se sugiere avanzar sobre el proyecto desarrollado en la Práctica Profesionalizante I El/ los proyectos que se desarrollen formarán parte del portfolio que cada estudiante generará como muestra de sus competencias a la hora de egresar y ejercer la profesión.

Los objetivos de aprendizajes a tener en cuenta para la evaluación al finalizar el cursado del módulo de la práctica profesionalizante serán:

- Presentar un proyecto que cumpla con las definiciones y requerimientos planteados por el cliente
- Implementar el sistema propuesto para verificar su funcionamiento y la mínima cantidad de errores en relación a las pruebas realizadas (testeo).

## **1ER CUATRIMESTRE**

### **21. Gestión de proyectos**

Referencia a funciones del perfil profesional del Técnico Superior en Desarrollo de Software

- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto

esto comprende:

Obtener métricas a partir de los registros de actividades.

Reportar avances y dificultades.

Planificar sus actividades.

Controlar sus actividades.

Coordinar actividades de acuerdo a roles y funciones del equipo de trabajo.

Elaborar interrogantes coherentes al problema o proyecto planteado

Escuchar y reformular conceptos en interrelación dialogada.

Recibir indicaciones

Reconocer sus potencialidades personales y ajenas al momento de integrar equipos de trabajo

Jerarquizar metas y determinar secuencias y plazos de realización

<b>Capacidades a desarrollar</b>	<b>Contenidos</b>
<ul style="list-style-type: none"><li>• Conocer y utilizar técnicas y herramientas de relevamiento y</li></ul>	Concepto de proyecto. Elementos de Gestión. Etapas y criterios para la

<p>análisis de necesidades de información de distintos usuarios/clientes</p> <ul style="list-style-type: none"> <li>● Conocer y utilizar herramientas y métodos de planificación de proyectos, y de organización y distribución de tareas en equipos de trabajo</li> <li>● Identificar e integrar en la planificación de proyectos criterios de factibilidad económica, legales y éticos</li> </ul>	<p>planificación de Proyectos. Campos de aplicación. La producción por proyectos. Los proyectos en las organizaciones. Enfoque de gestión de proyectos. Etapas en la gestión de un proyecto. Métodos de planificación de proyectos. Métodos PERT/CPM. Diagramas de redes. Concepto de camino crítico. Diagramas temporales de planificación de proyectos. Diagramas de Gantt.</p> <p>Gestión de Proyecto de software: herramientas. Estimaciones. Gestión de Proyectos: conceptos fundamentales. Software de aplicación. Desarrollo del plan de proyecto.</p> <p>Gestión de Riesgos. Conceptos. Tipos de riesgos. Definición de riesgos. Clasificación según impacto. Plan de mitigación de riesgos.</p> <p>Gestión de la Calidad del Software. Conceptos. Visiones. Aseguramiento de la Calidad. Normas existentes. Validación y verificación.</p> <p>Gestión del Mantenimiento. Conceptos. Técnicas y Herramientas. Plan de mantenimiento de software.</p> <p>Gestión de las Configuraciones. Configuraciones del software: concepto. Gestión de las configuraciones: conceptos, etapas. Software de aplicación.</p> <p>Cuidado del ambiente, eficiencia energética y uso racional de los recursos naturales.</p>
---	---

## 2DO CUATRIMESTRE

### 22. Validación y Verificación de Programas

- Verificar la calidad de las aplicaciones, diseñando casos de prueba unitarios y de integración

esto comprende:

- Considerar las necesidades de cobertura de la prueba.
- Diseñar los casos de prueba.
- Preparar el entorno de pruebas.
- Realizar pruebas unitarias.

Registrar casos de prueba, datos y resultados de pruebas y acciones correctivas.

- Documentar en base a normas o estándares establecidos

esto comprende:

Registrar actividades realizadas.

Documentar todos los productos de su labor con los requerimientos funcionales y técnicos y de normas de calidad.

<b>Capacidades a desarrollar</b>	<b>Contenidos</b>
<ul style="list-style-type: none"><li>● Identificar los diferentes tipos de pruebas involucradas en la construcción del software.</li><li>● Identificar los diferentes roles y entregables generados a partir de la planificación y ejecución de las pruebas.</li><li>● Conocer y utilizar las técnicas de ejecución de pruebas</li></ul>	<p>Distinción entre validación y verificación. Enfoques estáticos y dinámicos. Fundamentos de “testing”, testeo de caja negra y de caja blanca. Pruebas funcionales: generación de casos o datos de prueba, clases de equivalencia. Pruebas estructurales: prueba estáticas, pruebas dinámicas, cobertura de la prueba. Otro tipo de objetivos: verificación de usabilidad, confiabilidad, seguridad.</p> <p>Registro de fallas, seguimiento de fallas e informes técnicos.</p> <p>Prueba unitaria, de integración, validación y prueba de sistema. Desarrollo conducido por el testeo. Refactorización del código. Testeo de regresión. Verificación y validación de artefactos que no constituyen código: documentación, archivo de ayuda, material de capacitación. Inspecciones, revisiones cruzadas, auditorías.</p> <p>Identificación de técnicas de prueba para desarrollar mejores casos de pruebas.</p> <p>Reporte y documentación de defectos con el propósito de identificar situaciones que pongan en riesgo los plazos de entrega de un producto. Investigación de Herramientas de automatización de pruebas que existen en el mercado. Confección de Plantillas de Reportes de Resultados de la aplicación de Técnicas de Caja Negra, de Caja Blanca y de Inspección del software</p>

## 11. PRÁCTICAS FORMATIVAS

Se establece que en cuanto al total de horas destinadas a prácticas formativas: "corresponde un mínimo del 33% de la carga horaria total de los campos involucrados y debe hallarse distribuida de manera equilibrada en todos los años de la trayectoria formativa". Las Prácticas formativas están destinadas a validar y perfeccionar saberes, habilidades y capacidades que deben alcanzar los estudiantes en relación a un espacio curricular específico o a un grupo articulado de ellos.

Podrán considerarse Prácticas Formativas para la formación del **Técnico Superior en Desarrollo de Software**

Campos de Formación	Se sugiere	Articulación
General	Estudio de casos relacionados a los espacios curriculares del campo de la formación de fundamento, de la formación específica y de la práctica profesionalizante y al perfil profesional, con complejidad gradual y creciente.	Abordaje transversal del currículum basados en propuestas auténticas que respondan a los intereses de los estudiantes y articulen todos los espacios curriculares del diseño, con foco en el alcance y el perfil profesional propuesto
Fundamento	Visitas a organizaciones, resolución de situaciones problemáticas, simulación, estudio de casos preferentemente en articulación con los otros campos de formación y orientados al perfil profesional	
Específico	<p>Estudio de situaciones reales o simuladas, resolución de situaciones problemáticas, ejercicios de análisis, ejercicios de optimización y/o modificación de desarrollos digitales, preferentemente en articulación con los otros campos de formación y orientados al perfil profesional que habiliten a:</p> <ul style="list-style-type: none"> <li>- Desarrollar una base de datos normalizada con la mayor cantidad de información para evitar inconsistencias a la hora de desarrollar el sistema.</li> <li>- Desarrollar un sistema con código reutilizable con el fin de optimizar el funcionamiento.</li> <li>- Realizar pruebas simulando fallas que puedan ocurrir durante la carga o consulta de datos.</li> <li>- Registrar decisiones de diseño, elementos utilizados y resultados de pruebas.</li> <li>- Desarrollar tienda virtual (carrito de compras) con CMS (Sistema de gestión de contenidos)</li> <li>- Escribir y reutilizar código para aplicaciones móviles.</li> <li>- Creación de interfaces móviles.</li> <li>- Prueba en emulador virtual y dispositivo físico.</li> <li>- Publicación en Store correspondiente.</li> </ul>	

## 11. CONDICIONES DE EGRESO

Haber cursado y aprobado todos los espacios curriculares incluidos en la estructura de la carrera.

## 12. PERFIL PROFESIONAL DEL DOCENTE

Las denominaciones académicas de titulación consignadas en el cuadro que a continuación se expone, son orientativas.

Para cubrir los Espacios Curriculares los postulantes deberán contar con titulación Superior<sup>2</sup> A los fines de la selección se deberá considerar el “Perfil Docente” como instancia prioritaria<sup>3</sup>

Espacios Curriculares	PERFIL DOCENTE <sup>4</sup>		Denominaciones Académicas posibles <sup>5</sup>
	Conocimiento en:	Con experiencia en:	
<ul style="list-style-type: none"> <li>- Elementos de matemática y lógica</li> <li>- Estadística y probabilidad aplicadas</li> </ul>	<ul style="list-style-type: none"> <li>- Uso de software específico</li> </ul>	<ul style="list-style-type: none"> <li>No excluyente</li> </ul>	<ul style="list-style-type: none"> <li>- Profesor de Matemáticas de Nivel superior</li> <li>- Licenciado en Matemáticas</li> <li>- Ingeniero o Licenciado en Informática</li> </ul>
<ul style="list-style-type: none"> <li>- Programación I</li> <li>- Base de datos</li> <li>- Programación II</li> <li>- Programación III</li> <li>- Práctica Profesionalizante I</li> <li>- Práctica Profesionalizante II</li> <li>- Interfaz de usuario</li> <li>- Verificación y Validación de programas</li> </ul>	<ul style="list-style-type: none"> <li>- Programación estructurada</li> <li>- Programación Orientada a Objetos</li> <li>- Lenguaje Orientado a Objetos (.NET, Java o Phyton)</li> <li>- Administración de motor de base de datos</li> <li>- HTML, CSS, JavaScript</li> <li>- Metodologías ágiles. UML.</li> <li>- Git y Repositorios remotos.</li> <li>- Metodologías ágiles.</li> <li>- Herramientas para la organización de equipos.</li> </ul>	<ul style="list-style-type: none"> <li>- Procesos de desarrollo de software</li> <li>- Programación experiencia y actuación profesional en la disciplina</li> <li>- Dictado de clases en el paradigma Orientado a Objetos</li> <li>- Diseño y gestión de bases de datos</li> <li>- Gestión de proyectos de software</li> </ul>	<ul style="list-style-type: none"> <li>- Ingeniero en Sistemas</li> <li>- Licenciado en Sistemas de Información</li> <li>- Analista de Sistemas</li> <li>- Técnico Superior en Desarrollo de Software</li> <li>- Técnico Superior en Desarrollo Web y Aplicaciones Digitales</li> </ul>
<ul style="list-style-type: none"> <li>- Sistemas y Organizaciones</li> <li>- Modelado y Arquitectura de Software</li> <li>- Ingeniería de Software</li> </ul>	<ul style="list-style-type: none"> <li>Procesos en las organizaciones</li> <li>Metodologías</li> </ul>	<ul style="list-style-type: none"> <li>- Procesos de desarrollo de software</li> <li>- Análisis funcional de procesos</li> </ul>	<ul style="list-style-type: none"> <li>- Ing. en Sistemas</li> <li>- Licenciado en Sistemas de Inf.</li> <li>- Analista de Sistemas</li> <li>- TS en Desarrollo de Software</li> </ul>

<sup>2</sup> Ley Nacional de Educación Superior N° 24.521.

<sup>3</sup> Valorar significativamente las acreditaciones que presenten los profesionales en formación docente.

<sup>4</sup> Se deberá valorar significativamente al profesional docente que más se acerque a la conjugación de los conocimientos y experiencia laboral descrita.

<sup>5</sup> Otras titulaciones con alcance para el nivel y específicos para el dictado del Espacio Curricular.



<ul style="list-style-type: none"> <li>- Aproximación al mundo del trabajo</li> <li>- Ética y deontología profesional</li> <li>- Gestión de proyectos</li> <li>- Competencias Comunicacionales I</li> <li>- Competencias Comunicacionales II</li> </ul>	Liderazgo, Gestión de proyectos, Aspectos éticos y legales informáticos	Desarrollo de emprendimientos productivos y tecnológicos. Experiencia y actuación profesional en la disciplina	<ul style="list-style-type: none"> <li>- Ingeniero en sistemas</li> <li>- Ingeniero en computación</li> <li>- Licenciado en sistemas de información</li> <li>- Analista de Sistemas</li> <li>- Técnico Superior en Desarrollo de Software</li> <li>- Técnico Superior de Desarrollo web y Aplicaciones digitales</li> <li>- Licenciado en Comunicación</li> </ul>
- Inglés	TIC (nuevas tecnologías)	El manejo del idioma en el campo específico.	<ul style="list-style-type: none"> <li>- Profesor de Inglés de Nivel superior</li> <li>- Licenciado en Inglés</li> <li>- Traductor de Inglés</li> </ul>
<ul style="list-style-type: none"> <li>- Arquitectura de las computadoras</li> <li>- Sistemas operativos</li> <li>- Redes</li> </ul>	Hardware y componentes	Instalación y administración de sistemas operativos y redes	<ul style="list-style-type: none"> <li>- Ingeniero en sistemas</li> <li>- Ingeniero en computación</li> <li>- Licenciado en sistemas de información</li> <li>- Analista de Sistemas</li> </ul>

### 13. CAMPOS DE FORMACIÓN. Según Resolución C.F.E N° 229/14 – Anexo I

AREA DE FORMACION	CURSO	HORAS RELOJ	HORAS TOTALES	Porcentaje
<b>FORMACIÓN GENERAL</b>				
Competencias Comunicacionales I	1ero, 1er C	3	171	10 %
Aproximación al mundo del trabajo	1ero, 1er C	3		
Competencias Comunicacionales II	1ero, 2do C	3		
Ética y deontología profesional	1ero, 2do C	3		
Gestión de proyectos	3ero	4		
<b>FORMACIÓN DE FUNDAMENTO</b>				
Elementos de matemática y lógica	1ero	3		
Sistemas y organizaciones	1ero	4		

Arquitectura de las computadoras	1ero, 2do C	4	427	26 %
Inglés	2do	4		
Estadística y probabilidad aplicadas	2do	3		
Sistemas operativos	2do, 1er C	4		
Redes	2do, 2do C	4		
<b>FORMACIÓN ESPECÍFICA</b>				
Programación I	1ero	5	705	44 %
Base de datos	1ero	4		
Modelado y Arquitectura de Software	2do	4		
Programación II	2do	5		
Interfaz de usuario	3ero	3		
Ingeniería de software	3ero	5		
Programación III	3ero	5		
Verificación y Validación de programas	3ero	4		
<b>PRÁCTICA PROFESIONALIZANTE</b>				
Práctica Profesionalizante I	2do	7	320	20%
Práctica Profesionalizante II	3ero	8		

## 14. ACREDITACIÓN DE SABERES DE TRAYECTORIAS FORMATIVAS DE OTROS ÁMBITOS

### DISPOSITIVO DE ACREDITACIÓN Y RECONOCIMIENTO DE SABERES

#### Introducción:

Esta propuesta de acreditación de saberes tiene sus fundamentos en las normativas educativas nacionales y provinciales.

Tanto en la Ley de Educación Nacional N° 26206 como en la Ley de Educación Técnico Profesional N° 26058, y en la Ley N° 9870 de la Provincia de Córdoba, se encuentran referencias directas a la acreditación, reconocimiento, certificación, articulación jurisdiccional y con otros sectores públicos y de la producción.

Considerando las referencias a la acreditación de saberes, la Ley de Educación Técnico Profesional N 26.058 dispone - entre otros lineamientos - favorecer el reconocimiento y certificación de saberes y capacidades, la reinserción voluntaria en la educación formal, la prosecución de estudios regulares en los diferentes niveles y modalidades del Sistema Educativo y propiciar la articulación entre los distintos ámbitos y niveles de la Educación Técnico-Profesional. Asimismo, la Resolución CFE No 295/16 regula la carga horaria de las

ofertas de carácter diversificado para ser acreditadas con saberes y/o prácticas relacionados al campo de la formación específica de la titulación. Por su parte la Resolución Ministerial de la jurisdicción, RM N° 412/2010 - Régimen Académico Marco - señala que los estudiantes de carreras de Formación Técnica, que hayan realizado una actividad laboral y /o hayan recibido capacitación, de acuerdo con espacios curriculares contemplados en los Planes de Estudio, podrán solicitar reconocimiento de sus saberes. Expresa además que la evaluación se realizará mediante una Comisión Evaluadora que considere una instancia escrita y una instancia oral, debiendo obtener una calificación mínima de (7) siete puntos. En este marco se define la propuesta de acreditación de saberes para la TS en Desarrollo de Software.

### **Fundamentos Y Objetivos**

Tal como lo expresa la Resolución CFE 344/18, la noción de acreditación refiere a un proceso pedagógico que da cuenta de lo que los sujetos saben para hacer lo que hacen, para luego poner en diálogo estos saberes con los que el propio sistema educativo transmite y certifica. Se trata de identificar, reconocer, validar y certificar oficialmente que una persona ha adquirido determinados aprendizajes, para hacerla acreedora de un certificado oficial de un saber general por parte del sistema educativo.

Con el objetivo de valorar las trayectorias educativas de otros ámbitos educativos tanto como la experiencia profesional adquirida en el ámbito laboral, resulta oportuno reconocer el trayecto de formación y los saberes que tengan pertinencia con el perfil de la Tecnicatura Superior en Desarrollo de Software posibilitando el acceso a una instancia que le permita dar cuenta de esos conocimientos para luego organizar una trayectoria diferente de cursada y acreditación de espacios curriculares, propiciando un recorrido más ágil para el egreso.

### **Acreditación de saberes para la Tecnicatura Superior en Desarrollo de Software**

Para los estudiantes que provengan del nivel educativo secundario y aquellos idóneos en el sector que presenten la acreditación de saberes se realizará a través de dos instancias: entrevista y evaluación oral y escrita.

### **Procedimiento**

1- Solicitud del estudiante mediante nota dirigida a la Dirección del Establecimiento en los plazos establecidos.

2- Presentación de documentación respaldatoria: títulos, certificaciones de capacitaciones profesionales realizadas, proyecto de la pasantía y/o actividades de la práctica profesionalizante realizadas, constancias de trabajo relativos a la acreditación y otros que se consideren pertinentes.

3- Se conformará una Comisión ad hoc integrada por un miembro del equipo directivo (o quién éste designe), referente de la carrera, docente/s del o los EC a considerar para la acreditación.

4- La comisión ad hoc tendrá a su cargo las siguientes funciones y/o actividades:

- Analizar y valorar la documentación presentada por los/las estudiantes peticionantes
- Entrevistar y orientar a los interesados cuya documentación cumpla con los requisitos preestablecidos. Considerar en esta etapa aspectos que resulten relevantes para el procedimiento de acreditación como un proceso de reflexión en el

que se propicia la revisión crítica, la consolidación y el reconocimiento de los saberes previos.

- Si corresponde, la comisión informará al estudiante interesado el paso a la siguiente etapa de acreditación fijando la fecha y horario de la misma.
- La elaboración de el o los instrumentos de acreditación a partir de las funciones profesionales y de las actividades que estructuran los EC del trayecto seleccionado.
- Informar a los estudiantes inscriptos interesados las características y metodología de la instancia de acreditación y posibilitarles la práctica previa de procesos similares a los previstos.
- La acreditación consistirá en un momento de trabajo escrito donde se plantearán actividades prácticas consistentes en situaciones reales y propias del ejercicio profesional y una instancia de coloquio oral de defensa del trabajo escrito. Las actividades previstas en ambas instancias de acreditación contemplarán la resolución de situaciones problemáticas en las que se pongan en juego capacidades y competencias que den muestra de los saberes de orden técnico, así como también aspectos comunicativos y comportamentales relevantes para el ejercicio del rol profesional.
- Implementar la instancia de acreditación en las fechas acordadas debiendo el estudiante obtener una calificación mínima de siete (7) o más puntos, resultado que se computará para obtener el promedio de egreso

5- Finalizado el proceso, la Comisión ad hoc emitirá una resolución estableciendo los resultados de la acreditación y la consecuente nueva trayectoria formativa del solicitante.

## 16. CERTIFICACIONES INTERMEDIAS

Teniendo en cuenta la Resolución del Consejo Federal de Educación N° 295/16 “CRITERIOS PARA LA ORGANIZACIÓN INSTITUCIONAL Y LINEAMIENTOS PARA LA ORGANIZACIÓN DE LA OFERTA FORMATIVA PARA LA EDUCACIÓN TÉCNICO PROFESIONAL DE NIVEL SUPERIOR” y, habiendo analizado y comparado los espacios curriculares del primer año del presente diseño curricular:

Programación I

Base de Datos

Sistemas y Organizaciones

Ética y Deontología Profesional

con el trayecto formativo propuesto en la **Resolución CFE N° 289/16**, corresponde certificar la formación profesional de **Programador**.

De igual modo, habiendo analizado y comparado los espacios curriculares del tercer año:

Interfaz de usuario

Programación III

con el trayecto formativo propuesto en la **Resolución CFE N° 351/19 Anexión I**, corresponder certificar la formación profesional de **Programador Web**. Se sugiere organizar

los contenidos dentro de los espacios curriculares mencionados para que la acreditación pueda concretarse finalizado el primer cuatrimestre.

**17. CRITERIOS ORIENTATIVOS PARA LA PRESENTACIÓN JURISDICCIONAL Y LA EVALUACIÓN DE SOLICITUDES DE VALIDEZ NACIONAL DE TÍTULOS Y CERTIFICADOS DE ESTUDIOS – LEY N° 26.206. R SICE 170/18 Anexo I Grilla síntesis del diseño curricular presentado:**

**Componentes curriculares**

**Respecto del referencial al perfil profesional:**

	Indicar la presencia	
a. Alcance del Perfil Profesional	Si	
b. Funciones que ejerce el profesional	Si	
c. Área ocupacional	Si	
d. Habilitaciones profesionales	Si	

Respecto de la trayectoria formativa: El diseño de la trayectoria según los lineamientos de las Res. CFE Nro.295/16 y 229/14. Se extraerá del diseño la siguiente información:

**Superior**

Para el formato o variante de diversificación:

Campo formativo	Carga horaria en horas reloj para cada campo
Formación general	171
Formación de fundamento	427
Formación específica	705
Prácticas Profesionalizantes	320
<b>Carga Horaria (Mayor o igual a 1400 hs. reloj)</b>	<b>1623</b>