

DISEÑO CURRICULAR

TECNICATURA SUPERIOR EN DESARROLLO DE SOFTWARE



MINISTERIO DE
**EDUCACIÓN, CULTURA,
CIENCIA Y TECNOLOGÍA**
SUBSECRETARÍA DE EDUCACIÓN
DIRECCIÓN DE EDUCACIÓN SUPERIOR

1. Oferta educativa cuya solicitud de registro y acreditación se presenta.

Datos Generales

Carrera: Tecnicatura Superior en Desarrollo de Software

Título: Técnico Superior en Desarrollo de Software

Sector/es de actividad socio productiva: Informática

Denominación del perfil profesional: Desarrollador de Software

Familia profesional: Informática

Nivel: Superior

Modalidad: Educación Técnico Profesional

Duración de la carrera: Tres (3) años

Condiciones de ingreso: Nivel Medio/Educación Polimodal/ Educación Secundaria completa.

2. Fundamentos

En el Libro Blanco de la Prospectiva TIC proyecto 2020 del Ministerio de Ciencia, Tecnología e Innovación Productiva de la Nación Argentina, encontramos:

“Dos fuerzas directrices fundamentales predominan y parecen determinar el futuro en el mundo actual. La primera es la creciente movilidad de todo: dinero, capital, bienes, personas, valores, cultura, ideas, fluyen a través de las fronteras nacionales y regionales con direcciones y velocidades nunca antes conocidas. La segunda es la poderosa interdependencia de las distintas partes del mundo, su creciente interacción y cooperación en economía, producción, desarrollo social, comunicaciones e intercambio humano. La globalización actual está fundamentalmente orientada por el flujo de capitales. Numerosos autores sugieren que las economías nacionales se encaminan al colapso dando lugar a un nuevo sistema mundial dirigido por las inversiones (o mejor, por los inversores). En las antiguas y poderosas economías industriales la manufactura tradicional solamente da cuenta por una pequeña parte de la producción económica y el empleo, al mismo tiempo que el papel de los servicios se expande de manera asombrosa.

El futuro de las personas, los países y las regiones depende crecientemente de sus conocimientos, creatividad, innovatividad y de la capacidad de adquirir y compartir estos intangibles en redes de todo tipo, de alcance global.” Cuatro factores aparecen en el horizonte como las “claves del éxito” en el mundo global. Abordarlos, comprenderlos profundamente y desarrollar políticas y estrategias

adecuadas en los contextos locales y globales, es el único camino viable hacia el progreso y el bienestar. Estos factores son:

- Desarrollar una fuerza de trabajo educada y competente
- Dominar el proceso de innovación
- Desarrollar nuevos servicios para el mercado global
- Aprovechar el conocimiento global

En la República Argentina en el año 2003 se sanciona la Ley N° 25.856 que en su Artículo 1° establece que la actividad de producción de software "debe considerarse como una actividad productiva de transformación asimilable a una actividad industrial a los efectos de la percepción de los beneficios impositivos, crediticios y de cualquier otro tipo que se fijen para la industria por parte del Gobierno Nacional".

Posteriormente, en el año 2011 se sanciona la Ley N° 25.922 de promoción de la Industria del Software para todo el territorio argentino, estableciendo un régimen de estabilidad fiscal especial, a fin de fomentar esta industria.

Por su parte, el Instituto Nacional de educación Tecnológica (INET), promueve la mejora continua de la calidad de la Educación Técnico Profesional, asegurando mayores niveles de inclusión y adecuando en forma permanente la oferta educativa a las necesidades sociales, productivas y territoriales, para lo cual existen los siguientes regímenes regulatorios:

- Los Marcos de Referencia para las carreras del campo de la informática, en vistas al posterior proceso de homologación de títulos, según lo previsto en la Resolución N° 261/06 del Consejo Federal de Cultura y Educación (CFCyE).
- El Marco de Referencia para el Perfil Profesional correspondiente al título de Técnico Superior en Desarrollo de Software, construido por este grupo de trabajo, se aprueba en julio de 2011 mediante Resolución N° 129/11 del Consejo Federal de Educación (CFE), Anexo V.

La propuesta se encuadra en el marco de la Ley de Educación Nacional N° 26206/06, la Ley de Educación Superior N° 24521/95 y la Ley de educación Provincial 6691/10.

Se plantea al sistema educativo el desafío de formar recursos humanos que puedan dar respuestas a estas nuevas y cambiantes demandas, la necesidad de desarrolladores de software argentinos realicen trabajos para otros países, trabajos "offshore", residiendo en nuestro país. Las exportaciones de software

3

que realiza la República Argentina se incrementaron en los últimos años, pero sus posibilidades se han visto limitadas debido a la falta de profesionales para satisfacer estas demandas, transformándose esta situación en una problemática estructural del sector.

En ese sentido, las competencias profesionales de los trabajadores de este sector se han ido transformando y actualmente se exigen perfiles laborales con

formaciones más específicas, en permanente actualización. Esto genera una inmediata repercusión en las competencias requeridas en el mercado laboral, generando nuevos espacios de empleabilidad, demandando perfiles profesionales capaces de responder a las actuales necesidades de información de los distintos ámbitos del trabajo y de la producción.

El presente Plan de Estudios de la "Tecnatura Superior en Desarrollo de Software" propone un trayecto formativo que permitirá a los futuros técnicos, egresados de esta carrera, desempeñarse en el diversos ámbitos de trabajo en el diseño, construcción y verificación de artefactos de software, respondiendo a las necesidades locales y regionales, en el marco de las políticas de planeamiento estratégico provincial y nacional.

3. Objetivos Generales

- Formar profesionales capaces de realizar programas o componentes de sistemas informáticos con técnicas avanzadas de desarrollo, interpretar especificaciones de diseño, documentar productos informáticos, verificar y validar componentes de software, y crear, corregir o adaptar programas de acuerdo a los requerimientos de cualquier organización pública o privada, corporativa, pequeñas y medianas empresas o doméstica, a fin de responder a demandas y necesidades socio productivas locales y regionales vinculadas con el sector de la informática, software y servicios, en el marco del plan de desarrollo estratégico provincial y nacional.
- Formar recursos humanos dentro del campo de la Informática comprometidos con su contexto, con espíritu crítico frente a potenciales riesgos que conllevan las nuevas tecnologías, favoreciendo el acceso democrático, responsable y tolerante a las distintas fuentes de información.

4. Perfil Profesional

El perfil Profesional expresa el conjunto de realizaciones profesionales que una persona hace efectivas en las diversas situaciones de trabajo a las que puede enfrentarse en el dominio de su área ocupacional. A continuación se

4

presentan funciones y sub-funciones del perfil profesional de este técnico superior en las cuales se pueden identificar las siguientes actividades:

- Modelizar artefactos de software a partir de especificaciones, refinándolas en caso necesario, para determinar el diseño detallado y las características de una solución que las satisfaga en el contexto de la arquitectura del sistema de software del cual van a formar parte.

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.

Para realizar lo antes mencionado, el técnico utiliza lenguajes y herramientas de representación y modelización de sistemas, como UML y otras técnicas de graficación y especificación, incluyendo diccionarios de datos del proyecto y catálogos de patrones de diseño. También considera las características de la tecnología a utilizar y consulta a pares y al líder del equipo de trabajo para interpretar los problemas a resolver y verificar sus conclusiones y enfoques. Al realizar esto procura atenerse a los lineamientos de la arquitectura establecida para el proyecto y respeta criterios de seguridad informática, confidencialidad y las políticas vigentes en la organización en la cual se desempeña, así como las prácticas establecidas para el proyecto.

· Construir los artefactos de software que implementen el diseño realizado, aplicando patrones o reutilizando código en la medida en que resulte posible. Al hacer esto, y en función de lo acordado para el proyecto, optimizará el desempeño de lo que construya aplicando buenas prácticas de programación y documentación.

Esto comprende:

- Reutilizar elementos ya existentes.
- Redactar y elaborar códigos.
- Optimizar el código.
- Controlar cambios y versiones.
- utilizar ambientes de desarrollo.

Para realizar esto el técnico utiliza patrones, reutiliza código existente adaptándolo o complementándolo a su nueva función o redacta código nuevo

5

aplicando sus conocimientos de programación, respetando buenas prácticas y las normas establecidas para asegurar la calidad del proyecto. Esto implica el dominio del lenguaje y del ambiente de desarrollo utilizados en el proyecto, así como la tecnología en la cual va a ser implementada la solución. También consulta a pares y al líder del equipo de trabajo para reflexionar y recibir ayuda que le permita resolver problemas encontrados o aporta sus conocimientos y capacidad de reflexión a otros, y participa de foros y listas temáticas para encontrar soluciones o elementos reutilizables.

· Verificar los artefactos de software construidos considerando las necesidades de cobertura de la prueba. Para ello diseña los casos considerando el entorno de pruebas y ejecuta pruebas unitarias, así como registra los datos y resultados. De

ser necesario, realiza acciones correctivas sobre el código hasta satisfacerse de que cumpla con las especificaciones recibidas.

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.

Para realizar esto el técnico determina las necesidades de cobertura en función de las características de su asignación y normas establecidas para asegurar la calidad del proyecto, identifica las clases de equivalencia de datos utilizados internamente o intercambiados y diseña los casos de prueba, tomando en cuenta la estructura del artefacto y las condiciones de borde, así como prepara el entorno de pruebas, incluyendo los scripts y datos necesarios. Esto implica el dominio de conceptos de “testing” y de herramientas utilizadas para establecer el ambiente de “testing”. Realiza las pruebas unitarias, registrando los datos y resultados alcanzados, así como las acciones correctivas realizadas para solucionar las fallas encontradas.

· Revisar el código de artefactos de software para resolver defectos o mejorarlo. Este código puede ser propio o ajeno. Realizar revisiones cruzadas con otros integrantes del proyecto para asegurar la calidad del producto. Algunas asignaciones requieren una revisión de código ya existente para poder ampliar funcionalidades o refactorizarlo.

Esto comprende:

- Interpretar código.
- Diagnosticar defectos.
- Depurar defectos.

6

Al realizar esto el técnico analiza sistemáticamente el código para identificar partes relacionadas con posibles funcionamientos anómalos y revisa meticulosamente esas partes para determinar las causas de posibles defectos a fin de corregirlos, así como replantea, si resulta necesario, aspectos estructurales y cuida de no introducir otros defectos al efectuar modificaciones en el código. También analiza tanto el cumplimiento de buenas prácticas de programación, como la eficiencia del código.

· Documentar sus actividades y los resultados obtenidos aportando elementos para asegurar la calidad de los proyectos de acuerdo a normas y estándares establecidos.

Esto comprende:

- Registrar actividades realizadas.
- Documentar todos los productos de su labor.

Las normas de calidad del proceso de desarrollo de software exigen una adecuada documentación del mismo, así como del producto resultante. Para que el técnico pueda realizar un aporte efectivo a estas exigencias de calidad y para facilitar el mantenimiento de lo que desarrolle, tiene que justificar las decisiones relevantes de diseño que tome, así como las limitaciones que tienen los artefactos que produzca de acuerdo a criterios de legibilidad por parte de otros y a las normas de documentación establecidas para el proyecto.

· Gestionar sus propias actividades dentro del equipo de trabajo del proyecto. Realizar la planificación (organización y control) de las tareas a realizar, el oportuno reporte de avances y dificultades y el registro y reflexión sobre lo realizado para capitalizar experiencias y estimar métricas aplicables a su actividad.

Esto comprende:

- Obtener métricas a partir de los registros de actividades.
- Reportar avances y dificultades.
- Planificar las actividades.
- Controlar las actividades.

La construcción de software es una actividad que se desarrolla por proyectos, los que son llevados a cabo por un equipo de trabajo y el técnico tiene que realizar un aporte efectivo al trabajo conjunto. Para ello tiene que mantener una comunicación efectiva con quien lidere el grupo o lo asesore y con el resto de su equipo de trabajo, informando y consultando sobre problemas que observe al enfrentar sus asignaciones. También debe desarrollar su propia profesionalidad estimando tiempos y comparando resultados, extrayendo conclusiones formales

7

o informales que le permitan establecer sus propias métricas de rendimiento y calidad, así como un estrecho autocontrol que le facilite una mayor predictibilidad de sus resultados.

Para lograr un desempeño competente en sus actividades profesionales, el desarrollador de software, además de realizar las actividades previstas en su perfil profesional e incluidas aquí en la descripción de las funciones que realiza, tiene que dominar ciertos aspectos de la tecnología de la información que le sirven de base para poder desarrollar competentemente sus funciones profesionales. Al dominio de estos aspectos lo hemos denominado: Desempeño de base.

Esto comprende:

- Conocer y saber utilizar con propiedad y en condiciones de seguridad recursos de hardware, software y redes para emplear los ambientes

que necesite para el desarrollo y la verificación del software, mantener los repositorios de información que necesite utilizar y disponer de los productos de su trabajo en condiciones de confiabilidad.

5. Habilitaciones Profesionales

Se establecen las siguientes habilitaciones profesionales. Éstas tienen efecto para el desempeño del profesional en forma autónoma o asumiendo la responsabilidad por los resultados que obtenga su grupo de trabajo:

- Diseñar, construir y verificar artefactos de software de complejidad media, correspondiente a sistemas de información o vinculados indirectamente al hardware o a sistemas de comunicación de datos, respondiendo a especificaciones.
- Controlar la calidad de artefactos de software para resolver defectos o mejorarlos, lo que incluye revisar especificaciones, diseño y código. Bajo la supervisión de equipos de desarrollo de software.
- Queda excluido de esta habilitación el software correspondiente a sistemas críticos para la seguridad como es el caso de los que involucran el procesamiento de información que conlleve riesgos efectivos para terceros. Queda excluido el software destinado a:
 - ✓ Control de equipos y procesos médicos, industriales o de domótica que puedan poner en riesgo inmediato o mediato la salud de personas.
 - ✓ Procesamiento de información crítica para los individuos: corroborar identidad, estado de salud, demostrar situaciones legales, fiscal, patrimonial u otras que afecten a su patrimonio o libertades.
 - ✓ Procesamiento en línea de transacciones financieras importantes.

8

6. Área Ocupacional

Este técnico se puede desempeñar 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 otras actividades, pero que producen el software que necesitan para desarrollar sus propias actividades o que integran en productos que venden.

Integra equipos de proyecto dedicados al desarrollo o mantenimiento de software, produciendo artefactos que satisfagan especificaciones y se integren al sistema objeto del proyecto.

Resuelve estas asignaciones individualmente o trabajando entre pares, recibiendo la supervisión y asesoramiento de un líder de proyecto o de grupo.

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, es posible una evolución ocupacional del Técnico Superior en Desarrollo de Software para 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, los Técnicos Superiores en Desarrollo de Software o profesionales equivalentes con capacidad emprendedora pueden y suelen asociarse entre ellos para generar sus propias empresas brindando servicios de desarrollo de software a terceros.

7. Plan de Estudios – Estructura Curricular

Primer Año

1 Inglés
2 Matemática Aplicada
3 Estructura de las Organizaciones
4 Tecnología de la Información
5 Arquitectura de Computadoras
6 Programación I
7 Bases de Datos
8 Taller : Informática Aplicada

Segundo Año

9 Realidad Socio-productiva y laboral de la Argentina
10 Dirección y Gestión de Proyectos Informáticos
11 Inglés Técnico
12 Sistemas Operativos y Redes
13 Análisis y Diseño de Sistemas I
14 Programación II
15 Programación de Sitio Web
16 Práctica Profesionalizante I

Tercer Año

17 Ética y Deontología Profesional
18 Legislación Informática
19 Ingeniería de Software
20 Análisis y Diseño de Sistemas II
21 Programación III
22 Interfaz Usuario
23 Práctica Profesionalizante II

10

Organización de Asignaturas por año y carga horaria.

Años	Orden	Asignaturas	Régimen	Horas Cátedras	
				Semanales	Totales
1er	1	Inglés	Anual	3	96
	2	Matemática Aplicada	Anual	4	128
	3	Estructura de las Organizaciones	Anual	4	128
	4	Tecnología de la Información	Anual	3	96
	5	Arquitectura de Computadoras	Anual	4	128
	6	Programación I	Anual	4	128
	7	Bases de Datos	Anual	4	128

	8	Taller : Informática Aplicada	Anual	4	128
	Total de Hs.Primer Año			30	960
2do	9	Realidad Socio-productiva y laboral de la Argentina	Anual	2	64
	10	Dirección y Gestión de Proyectos Informáticos	Anual	4	128
	11	Ingles Técnico	Anual	3	96
	12	Sistemas Operativos y Redes	Anual	6	192
	13	Análisis y Diseño de Sistemas I	Anual	4	128
	14	Programación II	Anual	4	128
	15	Programación de Sitio Web	Anual	3	96
	16	Práctica Profesionalizante I	Anual	6	192
		Total de Hs.Segundo Año			32
3er.	17	Ética y Deontología Profesional	Anual	3	96
	18	Legislación Informática	Anual	3	96
	19	Ingeniería de Software	Anual	4	128
	10	Análisis y Diseño de Sistemas II	Anual	4	128
	21	Programación III	Anual	4	128
	22	Interfaz Usuario	Anual	4	128
	23	Práctica Profesionalizante II	Anual	6	192
		Total de Hs.Tercer Año			28
TOTAL DE HORAS CATEDRAS DE LA CARRERA				90	2880
TOTAL HORAS RELOJ DE LA CARRERA				60	1920

Distribución de Asignaturas por Campos de Formación.

Campo de	Asignaturas	Horas Cátedras		Porcentajes
		Semanales	Totales	
Formación	Ingles	3	96	
General	Realidad Socioproductiva y laboral de la Argentina	2	64	9%

	Ética y Deontología Profesional	3	96	
Formación de Fundamento	Matemática Aplicada	4	128	24%
	Estructura de las Organizaciones	4	128	
	Tecnología de la Información	3	96	
	Dirección y Gestión de Proyectos Informáticos	4	128	
	Ingles Técnico	3	96	
	Legislación Informática	3	96	
Formación Específica	Arquitectura de Computadoras	4	128	54%
	Programación I	4	128	
	Bases de Datos	4	128	
	Taller : Informática Aplicada	4	128	
	Sistemas Operativos y Redes	6	192	
	Análisis y Diseño de Sistemas I	4	128	
	Programación II	4	128	
	Programación de Sitio Web	3	96	
	Ingeniería de Software	4	128	
	Análisis y Diseño de Sistemas II	4	128	
	Programación III	4	128	
	Interfaz Usuario	4	128	
Práctica Profesionalizante	Práctica Profesionalizante I	6	192	13%
	Práctica Profesionalizante II	6	192	

Practica Formativa

Coordinación por año	Hs. Semanales
1er.año	8
2do.año	8
3er.año	8
Total de hs articulas Semanales	24
Total de hs articulas Anuales	768

ASIGNATURAS**PRIMER AÑO****1. Ingles****Carga horaria**

Horas semanales	Horas anuales	Régimen
3	96	Anual

Contenidos Mínimos

Textos de estructura instructiva: instructivos y manuales de usuario. Exploración visual del texto. Estructura: títulos, subtítulos, imágenes visuales, párrafos, tipografía. Secciones. Reconocimiento de cognados. Las especificaciones. Reconocimiento de instrucciones. El imperativo (afirmativo y negativo). Instrucciones con verbos de carácter negativo. Propósito, condición y especificación de tiempo en la instrucción. Precauciones. Formación de palabras: sufijos afijos. Textos de estructura expositiva. La función informativa, la trama en que están estructurados. Texto expositivo con trama narrativa: procesos históricos o procesos naturales. Los textos expositivos de trama descriptiva y los textos clasificatorios. La organización del libro de texto académico. Sus partes. El capítulo: estructura. El párrafo: oración tópica. Las oraciones enunciativas. Uso de la tercer persona y la pasiva. Los verbos de las ideas principales en presente. Registro formal. El uso de términos técnicos o científicos. Relaciones semánticas. Formación de palabras: sufijos afijos. Conectores.

2. Matemáticas Aplicada**Carga horaria**

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos Mínimos

Conjuntos, diagramas de Venn, operaciones, complementos, producto cartesiano, radicación, potencia. Relaciones, tipos: reflexividad, simetría, transitividad, equivalencia Numerabilidad y cardinalidad. Aritmética modular. Relaciones de congruencia. Sistemas de numeración-Funciones; tipos: inyectivas, sobreyectivas, inversas,

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. 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.

Lógica: Elementos de lógica. Lógica proposicional, conectivos lógicos. Tablas de verdad. Formas normales; conjuntiva y disyuntiva. Validez. Lógica de predicados; cuantificadores universal y existencial. 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, contraejemplo, por contradicción. Inducción matemática. Inducción fuerte. Definiciones matemáticas recursivas. Buen ordenamiento.

Estadística descriptiva e inductiva. Variables. Distribución de frecuencia. Intervalos y límites de clase. Histogramas y polígonos de frecuencia. Curvas de frecuencia. Ojivas. Promedios y medidas de dispersión. Medidas de centralización. Nociones de probabilidad. Probabilidades. Variación simple y con repetición. Permutaciones simples y con repetición. Combinaciones simples y con repetición. Población y muestra. Teoría del muestreo. Selección de muestra al azar. Tabla de números aleatorios. Números índice. Concepto y aplicaciones. Presos relativos. Cantidad o volumen relativo. Valor relativo. Series cronológicas. Movimiento y análisis de las series de tiempo. Estimación de tendencia. Índices de variaciones estacionales. Desestacionalización de series. Deflación de series. Comparación de datos. Tipos de costos. Presupuestos.

3. Estructura de las Organizaciones

Carga Horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos mínimos

Tipos de organización. Enfoque sistémico de la organización. Organización: elementos, estructura y comportamiento Cultura, Clima, Eficacia en la organización. Diseño organizacional. Organización estructurada por o líneas de productos. Transferencia de la información. Nociones de procesos de gestión y transformación de materiales y su organización. Concepto de proceso. Procesos de ciclo de ventas y cobranzas; del ciclo de compras y ventas. Organización y procesos de servicios. Planificación estratégica. Niveles. Sistemas, y Equipos de información. Sistemas basados en la informática. Función de las computadoras en un sistema de información.

Práctica Formativa

Taller de Comunicación inter-organizacional. El acto de comunicar. El estilo de la organización. El estilo del gerente. El estilo del personal. La comunicación como sistema organizacional. Transmisión y eficacia. Comunicación y poder. Vinculo entre las variables. Texto, contexto y paratexto intertextualidad. Los géneros informativos en la comunicación de masas. La oralidad del lenguaje. La palabra articulada como poder y acción. Condiciones de la oralidad. Culturas narrativas y orales. El orador. Actitud del orador. Plan del discurso. Expresividad. Tono de voz. Volumen. La improvisación. La escritura. Coherencia y cohesión. Ortografía y gramática. Diferentes tipos de textos. Producción de textos.

4. Tecnología de la Información

Carga Horaria

Horas semanales	Horas anuales	Régimen
3	96	Anual

Contenidos Mínimos

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 y computación) o aportan a la misma. El problema de la complejidad. Concepto de computación paralela, concurrente, multi-núcleos. Evolución del computador, su organización y unidades funcionales que lo componen. Arquitectura interna de los computadores, unidad central de procesamiento, instrucciones y flujo de la información. Tipos y niveles de organización de la memoria interna y externa (sistemas de memoria, tecnologías y jerarquías, memoria caché, memoria virtual, dispositivos de almacenamiento secundario). Orígenes y evolución de Internet y las comunicaciones digitales. Arquitecturas de red. Especializaciones en la computación centrada en redes. Redes y protocolos. Computación distribuida. Paradigmas cliente/servidor y peer to peer. Computación sin cables y móvil.

5. Arquitectura de Computadoras

Carga Horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos Mínimos

15

Periféricos, clasificación y utilización. Funcionamiento del programa a nivel de la máquina. 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. Compresión de datos. Expresiones Lógicas y funciones de Boole. Representación de datos numéricos, aritmética binaria. El overflow. Representación de datos con punto flotante: rango, precisión y aritmética. Representación de caracteres, audio e imágenes. Lógica Combinacional. Compuertas lógicas. Dispositivos combinacionales. Lógica Secuencial. Flip Flop y sus aplicaciones. Arquitectura interna de la Computadoras. Hardware y, Software. Maquina de Von Neumann. Procesamiento Paralelo. Pipeline y Multi-núcleos. El modelo de Maquina Multinivel. Interpretación y Traducción. Unidad Central de Proceso. Instrucciones y flujo de información. El microprocesador. Memorias. Tipos y organización de las memorias. Memoria Jerárquica. Memoria Cache. Administración de la Memoria. Segmentación y Paginación. Memoria Virtual. Dispositivos de memoria. Periféricos: clasificación y utilización. Interrupciones. Interconexión: arquitectura del Bus. Programas a nivel de la maquina: ensambladores.

6. Programación I

Carga horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos Mínimos

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 e 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. Máquinas virtuales, concepto, jerarquía de máquinas virtuales, lenguajes intermedios, asuntos de seguridad que surgen al ejecutar código en una máquina diferente. Representación de datos numéricos, rango, precisión y errores de redondeo. Arreglos. Representación de datos de caracteres, listas y su procesamiento. Manejo de memoria en tiempo de ejecución, punteros y referencias, estructuras encadenadas, pilas, colas y tablas de hashing. Recolección de espacios no utilizados. La elección de una estructura de datos adecuada. Estándares de redes y cuerpos de estandarización. Modelos de referencia: modelo de capas, TCP/IP. Espacio de direcciones del protocolo,

categorías de direcciones. Máscara de red. Elementos de ruteo e interconexión. Aspectos de seguridad de redes.

Práctica Formativa.

Taller: Resolver ejercicios de programación, tanto con lápiz y papel como en el computador. Se espera que al concluir el ciclo los estudiantes se apropien significativamente de los paradigmas de programación de un lenguaje de alto nivel. Resolver diversas clases de problemas desarrollando capacidad de abstracción. Revisar y corregir programas dados. Resolver diversos tipos de problemas comenzando por especificar su propia comprensión de la asignación, diseñar una solución, programar o integrar partes de código utilizando el ambiente de programación indicado, documentación de acuerdo a buenas prácticas y realizar la verificación unitaria de lo realizado.

7. Bases de Datos

Carga Horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos Mínimos

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. Privacidad, integridad, seguridad y preservación de la información. La persistencia e integridad de los datos. Modelización de datos modelos conceptuales (E/R, UML), modelo orientado a objetos, modelo relacional, modelos estructurados (XML). Concepto y evolución de los sistemas de bases de datos, sus componentes, funciones de un sistema de base de datos. Lenguajes de consulta (SQL, QBE), definición de datos, álgebra relacional, formulación de consultas, lenguaje de actualización, restricciones, integridad. SQL embebido en un lenguaje imperativo. "Scripts". Introducción a un lenguaje de consulta de objetos. Procedimientos almacenados. Diseño de bases de datos, dependencia funcional, formas normales, descomposición de un esquema, claves primarias y secundarias. Procesamiento de transacciones, fallas y recuperación, control de concurrencia. Bases de datos distribuidas, problemas que surgen con su explotación.

Práctica Formativa.

Taller: Resolver ejercicios de álgebra relacional. Se espera que al concluir el ciclo los estudiantes resulten capaces de explotar una base de datos relacional. Revisar y corregir programas dados. Resolver diversos tipos de problemas comenzando por especificar consultas a bases de datos dadas, programar actualizaciones de datos

en base a cálculos con nuevos datos, preocupándose tanto por la integridad de la información como por la eficiencia de los procesos. Diseñar tablas y bases de datos, incorporar procedimientos. Desarrollar proyectos grupales durante los cuales se simulen condiciones similares a las del trabajo profesional y cada uno aporte componentes que deben integrarse en el producto final. Diseñar tablas y bases de datos relacionales, analizar y discutir su eficiencia escalabilidad. A partir de un diseño, analizar clases de equivalencia y diseñar esquemas de prueba.

8. Taller: Informática Aplicada

Carga Horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos Mínimos

Observación de sistemas externos o internos al establecimiento, reales o simulados. Reconocimiento de distintos tipos de software a partir de utilización y observación de programas. Reconocimiento de distintos tipos de hardware y su funcionalidad. Registración de observaciones, documentación de las actividades realizadas. Comprender el funcionamiento de los dispositivos observados.

SEGUNDO AÑO

9. Realidad Socio-productiva y laboral de la Argentina

Carga horaria

Horas semanales	Horas anuales	Régimen
2	64	Anual

Contenidos Mínimos

Transformaciones del Estado-Nación. Rol social del estado. Ciudadanía y espacio público. Problemáticas Socio Culturales. Proceso de globalización, transnacionalización y regionalización. Procesos políticos, económicos y su vinculación con el mundo del trabajo actual. La economía como dimensión de la Vida social. Trabajo y sociedad. Formas de organización del trabajo. Evolución de la organización social del trabajo. Modelos.

Mundo del trabajo, subjetividades e identidades colectivas. Particularidades del mercado de trabajo en Argentina. Medios masivos de comunicación. El conocimiento en la "sociedad de la información". Estratificación socioeconómica. Transformaciones del estado moderno. Estado y sociedad. Reforma del Estado Argentino. Rol del Estado.

Territorio, ambiente y problemáticas locales. De los estados nacionales a la Globalización y de ésta a la regionalización o a lo territorial. La integración en bloques regionales y una perspectiva latinoamericana. Globalización y regionalización. El Mercosur: trayectorias situación y propuestas. La Argentina en los nuevos escenarios internacionales. Modernidad y Postmodernidad y mundo del trabajo. Las transformaciones del Estado Contemporáneo. Desarrollo Local y Regional. Modelos de desarrollo local. Procesos de intervención. Etapas de un proyecto de desarrollo local. Planeamiento táctico y estratégico. Democracia y participación en el proceso de planeamiento. Análisis de casos.

Seminarios sobre las Interrelación de los principales problemas sociales de Argentina relacionados con la pobreza, el trabajo y la educación.

10. Dirección y Gestión de Proyecto Informáticos

Carga horaria

Horas semanales	Horas anuales	Régimen
3	96	Anual

Contenidos Mínimos

Gestión de equipos. Planificación de proyectos. Métricas del software y técnicas de estimación. Análisis y gestión de riesgos. Control de la calidad del software. Gestión de la configuración del software. Herramientas de gestión de proyectos. Introducción a la dirección de proyectos. Fases de un proyecto. Fase de planeación. Fase productiva. Recursos humanos. Software libre de gestión de proyectos

Taller: Casos de éxito. Casos de estudio. Herramientas de financiamiento de proyectos.

11. Inglés Técnico

Carga horaria

Horas semanales	Horas anuales	Régimen
3	96	Anual

Contenidos Mínimos

Las revistas de difusión tecnológica. Sus características: la heterogeneidad. Distintos tipos de códigos. Distintos niveles del lenguaje. Textos informativos de diversas clases, textos publicitarios, informativos y de opinión, etc. Los estilos y distintos géneros periodísticos. Los hechos, las opiniones y los argumentos. El pasado. La formulación de hipótesis. Formación de palabras: sufijos, afijos. Conectores. Las publicaciones en una investigación científica. Propósitos. Características. Partes: abstract: diferentes tipos. Elementos constitutivos. Patrones retóricos. Introducción: diferentes tipos. Elementos constitutivos. Patrones retóricos. Las citas (verbos evaluativos). El nicho "niche" Metodología: elementos constitutivos. Patrones retóricos. Resultados: patrones. Discusión y conclusiones: diferentes tipos. Elementos constitutivos. Patrones retóricos. Evidencia e interpretación. Aseveración vs. Hedging. Uso de conectores.

Redes Carga horaria

Horas semanales	Horas anuales	Régimen
6	192	Anual

Contenidos mínimos

Los sistemas operativos, su papel y propósito, la historia de su desarrollo, funcionalidades típicas. Mecanismos que soportan los modelos cliente-servidor y otros dispositivos. Características y objetivos de su diseño y su influencia en la seguridad, interoperabilidad, capacidad multimedia. Estructuras de sistemas operativos (monolíticas, modulares y de "micro kernel"). Abstracciones, procesos y recursos. Organización de los dispositivos, interrupciones: métodos e implementación. Concepto de estados 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. Ejecución concurrente: ventajas y desventajas. El problema de la exclusión mutua y algunas soluciones. Bloqueos: causas, condiciones, prevención. Paso de mensajes sincrónico y asincrónico. Problema de consumidor-productor y sincronización. Multiprocesamiento (interrupción de ciclos, reentrada). Políticas de despacho de procesos; programación con y sin prioridades de interrupción. Procesos e hilos de ejecución. Elementos de tiempo real y tiempos límite. Administración de memoria. Revisión de memoria física y hardware de administración de memoria. Paginación y memoria virtual. Set de instrucciones y cacheo de memoria. Administración de dispositivos, características de dispositivos seriales y paralelos. Abstracción de diferencias entre dispositivos. Estrategias de "buffering". Acceso directo a memoria. Recuperación de fallas. Seguridad y protección. Políticas y mecanismos de separación. Métodos y dispositivos de seguridad. Protección, control de acceso y autenticación. Backups. 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. "Scripting". Comandos básicos del sistema, creación de "scripts", pasaje de parámetros. Aspectos 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.

Redes de computadoras. Topologías. Clasificación. Protocolos. Servicios. Administración. Evolución Aplicaciones. Orientación. Topologías. Clasificación.

grupos. Usuarios. Seguridad en redes. Armado y configuración de redes.

Práctica Formativa

Taller: Localizar y selecciona artefactos de software, libre o bajo otras licencias, que respondan a ciertas características. Instalar, configurar y personalizar sistemas operativos, compiladores de lenguajes, editores y otros ambientes de programación o de prueba de programas. Crear y organizar repositorios de documentación y programas para uso.

13. Análisis y Diseño de Sistemas I

Carga horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos Mínimos

Análisis de requerimientos de software, el proceso, partes interesadas. Requerimientos funcionales, prioridad 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: diagramas de secuencia, máquinas de estado, diagramas de actividad. Pre y post condiciones. Principios generales de diseño: descomposición, desacoplamiento, cohesión, reusabilidad, portabilidad, testeabilidad, flexibilidad, escalabilidad. Patrones de diseño. Arquitecturas de software: concepto de vistas, arquitecturas distribuidas, "pipe-and-filter", "model-viewcontroller" (MVC). Diseño orientado a objetos. Diseño estructurado. Diseño orientado al recurso de componentes, incorporación de elementos disponibles al diseño. Diseño de interfaces con el usuario. Prototipos rápidos.

Práctica Formativa

Taller: Utilizar lenguajes y herramientas de representación y modelización de sistemas, como UML y otras técnicas de graficación y especificación, incluyendo diccionarios de datos del proyecto y catálogos de patrones de diseño. También considera las características de la tecnología a utilizar y consulta entre pares y líder del equipo de trabajo para interpretar los problemas a resolver y verificar sus conclusiones y enfoques.

14. Programación II

Carga horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos Mínimos

Diseño orientado a objetos, encapsulamiento y ocultamiento de información, separación entre comportamiento e implementación, clases y subclasses, herencia (sustitución), polimorfismo (subtipos vs. herencia), jerarquías de clases, clases colección y protocolos de iteración. Verificación unitaria d unidades de código, concepto de cubrimiento, organización, ejecución y documentación de la prueba.

Recursión, concepto de funciones matemáticas recursivas, funciones recursivas simples, estrategia de dividir y conquistar backtracking recursivo. Algoritmos de búsqueda sucesiva y binaria, de ordenamiento con tiempos cuadráticos (selección, inserción), quicksort, heapsort, mergesort. Tablas de hashing, estrategias para evitar colisiones. Árboles de búsqueda binaria. Representación de grafos. Algoritmos de camino mínimo. Concepto de autómata. Elementos de complejidad de algoritmos. Declaraciones. Modelos de declaración, elementos de verificación de tipos, tipos y polimorfismo paramétrico. Estándares de nomenclatura y formato en programación, encabezado de módulos u objetos con comentarios que expliciten sus alcances y limitaciones, inserción de comentarios o advertencias en el código, documentación adicional. Programación conducida por eventos, métodos para manejo de eventos, propagación de eventos, manejo de excepciones. Programación defensiva, importancia de verificar para evitar el overflow de arreglos y listas. T-SQL.

Práctica Formativa

Taller: Diseñar artefactos de software (clases, objetos, métodos, algoritmos, tablas) que resuelvan problemas planteados. Analizar críticamente la eficiencia y mantenibilidad de diseños alternativos. Relacionar situaciones con patrones de diseño. Analizar diversos tipos de arquitectura de sistemas de software, discutiendo sus propiedades de calidad (escalabilidad, portabilidad, seguridad, mantenibilidad). Construir prototipos rápidos con herramientas sencillas. Analizar y discutir su eficiencia y escalabilidad. A partir de un diseño, analizar clases de equivalencia y diseñar esquemas de prueba. Desarrollar proyectos grupales durante los cuales se simulen condiciones similares a las del trabajo profesional y en los que cada uno aporte componentes que deben integrarse en el producto final. Procesar pruebas e identificar defectos en artefactos producidos por otros. Planificar y diseñar casos y conjuntos de datos para prueba de artefactos dados, respondiendo a objetivos y requisitos de cobertura. Implementar pruebas de programas y pequeños sistemas utilizando herramientas y creando ambientes necesarios para realizar los procesos y revisar los resultados para generar informes de fallas.

Horas semanales	Horas anuales	Régimen
3	96	Anual

Contenidos Mínimos

Introducción general a la programación en la web, Web es un sistema Hipertexto/Hipermedia, Lenguaje HTML, Páginas estáticas Vs. dinámicas, Páginas dinámicas, Páginas dinámicas de cliente, Páginas dinámicas de servidor, Lenguajes web de cliente; Java script, Visual Basic Script, DHTML (HTML Dinámico), CSS, Applets de Java, Lenguajes 'web, de servidor; CGI, Common Gateway Interface; ASP, Active Server Pages; PHP, Hypertext. JSP, Java Server Pages; XML, Introducción a XML.

16.Práctica Profesionalizante I

Carga horaria

Horas semanales	Horas anuales	Régimen
6	192	Anual

Contenidos Mínimos

Colaboración en el desarrollo artefactos de software de baja y mediana complejidad dentro y fuera del Instituto, bajo supervisión. Configuración de hardware. Instalación y configuración de distintos tipos de software de mediana y alta complejidad. Documentar las actividades realizadas.

Propuestas formativas vinculadas a las actividades profesionales:

- Verificar los artefactos de software construidos considerando las necesidades de cobertura de la prueba. Interpretar y analizar críticamente determinaciones de diseño, sus características de la arquitectura del sistema de software existente. Detectar posibles lagunas o incoherencias, preguntando por aspectos faltantes o incoherentes, validando su propia interpretación conforme a procedimientos de calidad establecidos.
- Modelizar artefactos de software a partir de especificaciones, refinándolas en caso necesario, para determinar el diseño detallado y las características de una solución que las satisfaga en el contexto de la arquitectura del sistema de software del cual van a formar parte. Esto incluye depurar, revisar código propio o ajeno para optimizar funciones requeridas, conforme a los procedimientos de calidad establecidos.

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

Ello comprende la planificación (organización y control) de las tareas a realizar, el oportuno reporte de avances y dificultades y el registro y reflexión sobre lo realizado para capitalizar experiencias y estimar métricas aplicables a su actividad.

- Documentar sus actividades y los resultados obtenidos aportando elementos para asegurara la calidad de los proyectos de acuerdo a normas y estándares establecidos.

17. Ética y Deontología Profesional

Carga horaria

Horas semanales	Horas anuales	Régimen
3	96	Anual

Contenidos mínimos

El problema ético: Concepto de lo ético. Distinción entre Ética, Derecho, Psicología y Metafísica. Deontología o Ética Profesional. Los valores: Los valores morales. Posición subjetivista y objetivista frente a los valores. La norma moral y las escuelas éticas: Los actos humanos y los actos Hiel hombre. Significado de autonomía y heteronomía moral. Principales problemas morales do nuestro tiempo. La persona. Derechos. La actividad de la persona: conocer, obrar, hacer. El trabajo y la dignidad de la persona. La profesión y el enfoque ético. Perfeccionamiento propio y servicio social. Requisitos para el recto ejercicio de la profesión. Cualidades morales dg; la profesión. Código de Ética profesional. La propiedad intelectual y el acceso libre a la información y al conocimiento. Privacidad de datos personales. Normas que rigen el correo electrónico. Protección legal de la propiedad intelectual (incluyendo software), derechos de reproducción y derechos sobre marcas y patentes. Licencias de fabricación, de uso, GNU y "creative commons". 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 diviudos, normativa relativa a privacidad y "habeas data". Bases de datos públicas y privadas. Propiedad de datos empresarios. Secretos comerciales e industriales.

18. Legislación Informática

Carga horaria

Horas semanales	Horas anuales	Régimen
3	96	Anual

Contenidos mínimos

Derecho. Conceptos. Comienzo de la existencia física de las personas. Posturas antropológicas. Deberes y Derechos. La vida comunitaria. Límites. Conflicto. Proceso del conflicto. Mediación. El comportamiento social y el Medio ambiente.

Derecho del trabajo. Concepto. Denominaciones. Naturaleza y caracteres.

Contrato de trabajo. Concepto. Contrato de trabajo. El trabajo de la mujer.

Enfermedades y accidentes inculpables. Relaciones laborales. Concepto y objetivos.

Negociación colectiva. Convenios Colectivos de trabajo. Régimen legal.

Normativas acerca del ejercicio de la Profesión informática. Contexto normativo: Responsabilidades empresariales, contratos, responsabilidades del trabajador, leyes de protección de datos personales, propiedad intelectual de software, contenidos jurídicos aplicables a delitos informáticos. Distintas leyes de aplicación. Código de Ética. Código Penal: Ley de Propiedad Intelectual, derecho de reproducción y derechos sobre marcas. Licencias de fabricación, de uso, GNU y "creative commons". Legislación informática acerca de abuso de menores, sexualidad infantil y trata de personas. Valor de la información almacenada para las organizaciones, seguridad. Normativas sobre privacidad y "habeas data". Bases de datos públicas y privadas. Secretos comerciales e industriales. Normas que rigen el correo electrónico y redes sociales. Criterios de seguridad informática, confidencialidad y las políticas vigentes en la organización en la cual se desempeña.

Emprendedurismo. El contexto mundial y latinoamericano, ética y desarrollo, acción social y capital social, participación comunitaria, instrumentos técnicos para la gestión, alianzas estratégicas, instrumentos de monitoreo. Legislaciones específicas sobre las incumbencias profesionales del Técnico Superior.

19.Ingeniería de Software

Carga horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos mínimos

Conceptos de dinámica 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 comunicación, medies, ruidos que afectan a la comunicación. Importancia de la retroalimentación. La empatía. La escucha activa. Conflictos, técnicas de resolución alternativa. El equipo de proyectos de desarrollo de software, roles y responsabilidades de sus integrantes. Programas de trabajo y resolución conjunta de problemas. Modelos de ciclo de vida del software y de procesos de desarrollo. El problema del mantenimiento y las migraciones de plataforma. Metodologías tradicionales y ágiles. Metodologías ágiles, concepto de "sprint", fraccionamiento del producto en unidades realizables en un "sprint", cola de pendientes, mejora de productos provisorios. Documentación en el marco de un proceso en el que se aplican metodologías ágiles. Gestión de los cambios conceptos de versión. Interfaz

Herramientas de - versionado. Otras herramientas (bibliotecas, diccionarios, repositorios) del proyecto. El problema de asegurar la calidad y elementos de métricas. Modelos 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.

Práctica Formativa

Taller: Utilizar patrones, reutilizar código existente adaptándolo o complementándolo a su nueva función o redactar código nuevo aplicando sus conocimientos de programación, respetando buenas prácticas y las normas establecidas para asegurar la calidad del proyecto. Esto implica el dominio del lenguaje y del ambiente de desarrollo utilizados en el proyecto, así como la tecnología en la cual va a ser implementada la solución. También consultar a pares y al líder del equipo de trabajo para reflexionar y recibir ayuda que le permita resolver los problemas encontrados y aportar sus conocimientos y capacidad de reflexión a otros, y participar de foros y listas temáticas para encontrar soluciones o elementos reutilizables.

Proyectos de desarrollo de artefactos de software en los que se pongan en práctica diferentes metodologías. Poner en práctica estadísticas elementales propias del proceso personal de software. Realizar revisiones cruzadas de código proponiendo mejoras. Organizar la documentación de un proyecto. Utilizar herramientas de versionado y administración de la configuración. Reflexionar en forma conjunta sobre experiencias y conclusiones obtenidas.

20. Análisis y Diseño de Sistemas II

Carga horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos mínimos

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: diagramas de secuencia, máquinas de estado, diagramas de actividad. Análisis de datos: datos de referencia y de operaciones; datos de nivel de recursos y de volumen de actividad. Modelo Entidad / Relación (ER). Principales formas normales. Diccionario de datos. DER. Requerimientos no funcionales, ejemplos y su influencia en el diseño del

software. Herramientas de modelización. Validación de requerimientos. Estándares de documentos de requerimientos.

21.Programación III

Carga horaria

Horas semanales	Horas anuales	Régimen
4	128	Anual

Contenidos mínimos

Aplicaciones en redes. Protocolos a nivel de la capa de aplicación. Interfaces web: "browsers" y API. El "socket" como abstracción. Modelo cliente servidor. Primeras acciones de ambos. Creación de "sockets" y ligado de direcciones. Modelo cliente/servidor TCP. Concepto de sesión. Modelo cliente/servidor UDP. Concepto de paquete. "Polling" con primitivas bloqueantes. RCP. Tecnologías web, modelos de computación distribuida en la red. Protocolos web. Lenguajes de programación utilizadas para el desarrollo de páginas y sistemas web. Principios de ingeniería en la web. Sitios web estructurados mediante bases de datos. Tecnologías de búsqueda en web. El "middleware". 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. Intercambiar artefactos de software asumiendo la obligación de interpretar y criticar o mejorar lo recibido. Determinar las necesidades de cobertura en función de las características de su asignación y normas establecidas para asegurar la calidad del proyecto, identifica las clases de equivalencia de datos utilizados internamente o intercambiados y diseñar los casos de prueba, tomando en cuenta la estructura del artefacto y las condiciones de borde, así como prepara el entorno de pruebas, incluyendo los scripts y datos necesarios. Esto implica el dominio de conceptos de "testing" y de herramientas utilizadas para establecer el ambiente de "testing". Realiza las pruebas unitarias, registrando los datos y resultados alcanzados, así como las acciones correctivas realizadas para solucionar las fallas encontradas.

Práctica Formativa

Taller: Diseñar y programar aplicaciones sencillas que interactúen en un ambiente cliente-servidor. Diseñar sitios web organizados como bases de datos para que el usuario pueda actualizarlos sin intervención de desarrolladores. Utilizar ambientes de programación para web, programar aplicaciones interactivas que actualicen bases de datos, considerar y discutir aspectos de seguridad relativos a las mismas.

22.Interfaz Usuario

Carga horaria

Horas semanales	Horas anuales	Régimen
-----------------	---------------	---------

4	128	Anual
---	-----	-------

Contenidos mínimos

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. Proceso de desarrollo centrado en el usuario: foco temprano en los usuarios, prueba empírica de la calidad, diseño interactivo. Medidas de evaluación: utilidad, eficiencia, facilidad de aprendizaje, satisfacción del usuario. Modelos de diseño de la interacción: atención, movimiento, cognición, percepción y reconocimiento. 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 del usuario. Principios de interfaces gráficas, acción-objeto vs. objeto-acción, eventos en interfaces de usuario, estándares, errores más comunes. Interfaces para un sistema nativo, uso del browser. Sistemas que operan en la red. Considerar, discutir y diseñar interacciones software-usuario. Diseñar diversas pantallas que respondan a determinadas propuestas y evaluar conjuntamente lo obtenido. Diseñar interfaces para la web con ayuda: para la navegación. Diseñar interfaces para alguna norma estándar (USB, "bluetooth") para dispositivos.

23. Práctica profesionalizante II

Carga horaria

Horas semanales	Horas anuales	Régimen
6	192	Anual

Contenidos mínimos

Análisis y desarrollo de software de mediana y alta complejidad. Documentar el desarrollo, diseño e implementación del sistema. Actividades de gestión y autogestión de la actividad. Formular las especificaciones para que otros puedan trabajar en el equipo.

Propuestas formativas vinculadas a las actividades profesionales:

- Modelizar artefactos de software a partir de especificaciones, refinándolas en caso necesario, para determinar el diseño detallado y las características de una solución que las satisfaga en el contexto de la arquitectura del sistema de software del cual van a formar parte. Esto incluye depurar, revisar código propio o ajeno para optimizar funciones requeridas, conforme a los

procedimientos de calidad establecidos.

- Construir los artefactos de software que implementen el diseño realizado, aplicando patrones o reutilizando código en la medida en que resulte posible. Al hacer esto, y en función de lo acordado para el proyecto, optimizará el desempeño de lo que construye aplicando buenas prácticas de programación y documentación.
- Revisar el código de artefactos de software para resolver defectos o mejorarlo. Este código puede ser propio o ajeno. Esta actividad comprende revisiones cruzadas con otros integrantes del proyecto para asegurar la calidad del producto. Algunas asignaciones requieren una revisión de código ya existente para poder ampliar funcionalidades o refactorizarlo.
- Gestionar sus propias actividades dentro del equipo de trabajo del proyecto. Ello comprende la planificación (organización y control) de las tareas a realizar, el oportuno reporte de avances y dificultades y el registra y reflexión sobre lo realizado para capitalizar experiencias y estimar métricas aplicables a su actividad.
- Documentar sus actividades y los resultados obtenidos aportando elementos para asegurara la calidad de los proyectos de acuerdo a normas y estándares establecidos.

Bibliografía

Cabe mencionar que en los diferentes espacios curriculares, se trabajará con bibliografía y recursos educativos digitales actualizados debido a que tanto la tecnología como los perfiles profesionales evolucionan y cambian según la demanda ocupacional y el mercado de trabajo, por tanto son dinámicos y deben estar acorde al desarrollo tecnológico, social y mundial.