

UNIVERSIDAD NACIONAL DE HURLINGHAM

(Ley N° 27.016, sancionada el 19 de noviembre de 2014 y promulgada el 2 de diciembre de 2014)

Plan de Estudios: Licenciatura en Informática

Año: 2025

Universidad Nacional de Hurlingham

Licenciatura en Informática

1. Presentación

1.1. Denominación de la carrera:

Licenciatura en Informática

1.2. Títulos que otorga:

Intermedio: *Técnico/a Universitario/a en Programación*

Final: *Licenciado/a en informática*

1.3. Cantidad de horas de interacción pedagógica totales:

Titulación intermedia: 1.184 horas

Titulación Final: 3.200 horas

1.4. Cantidad de horas y créditos totales:

Titulación intermedia: 3.000 horas - 120 créditos

Titulación Final: 7.175 hs - 287 créditos

1.5. Modalidad de cursado:

Presencial

1.6. Institucionalidad de la carrera:

Instituto de Tecnología e Ingeniería

2. Fundamentación de la carrera

La UNAHUR es una universidad pública y gratuita que estructura su desarrollo académico y científico en base a cuatro ejes de estudio e investigación: salud, educación, tecnología e ingeniería y biotecnología. Tiene como misión contribuir a través de la producción y distribución equitativa de conocimientos e innovaciones científico-tecnológicas al desarrollo local y nacional, con un fuerte compromiso con la formación de excelencia y la inclusión al servicio del acceso, permanencia y promoción de sus estudiantes.

Esta misión, atenta a las demandas sociales y al desarrollo de la región, la calidad de vida y los valores democráticos, que valoriza los saberes de las comunidades locales, delinea un modelo de institución que refuerza el compromiso de la universidad para con su medio y, con ello, no subordina su labor a tareas solamente científico-tecnológicas sino que se asume como espacio de articulación entre el territorio y la institución universitaria que le pertenece.

La Universidad Nacional de Hurlingham se propone ofrecer una propuesta académica que permita atender las diferentes áreas vocacionales de sus potenciales estudiantes, sin perder de vista las necesidades locales de profesionales cualificados, a fin de asegurar tanto el desarrollo humano de su propia comunidad universitaria como de la local en su conjunto.

Desde la docencia se apuntará a brindar educación superior de calidad, formando profesionales de alto nivel, actualizados y en constante búsqueda de conocimientos, con un alto sentido ético-social.

Por otra parte, la investigación se nutrirá de las problemáticas docentes que se releven, así como de los núcleos de interés del estudiantado. El desarrollo industrial nacional necesita dotarse de recursos humanos altamente especializados que cubran los aspectos integrales del sector productivo, desde el conocimiento tecnológico específico hasta el inherente al planeamiento y gestión, considerando los aspectos de seguridad, éticos, sociales y ambientales, como la capacidad de generación de políticas públicas para el área.

El Instituto de Tecnología e Ingeniería de la Universidad Nacional de Hurlingham será el responsable de la transferencia de conocimiento necesaria para cubrir las vacancias del sector público y privado, y el escenario natural donde discutir la planificación estratégica de desarrollo tecnológico, incluyendo docencia, investigación y extensión.

El aumento sostenido que se espera en la demanda nacional y global de servicios asociados a las tecnologías de la información y las comunicaciones (TICs), necesita dotarse de recursos humanos altamente especializados que cubran los aspectos integrales del sector productivo en estas áreas. El país cuenta con varios de los factores necesarios para aprovechar este potencial. Particularmente en el ámbito de desarrollo de software, entre ellos una amplia base de empresas del sector de distintas características y tamaños que trabajan tanto en el mercado local como en

el internacional y una cantidad interesante de profesionales con capacidades competitivas a nivel global.

Es en este contexto que la Universidad Nacional de Hurlingham se propone aportar al tejido productivo local y al sector de las TICs nacional, recursos humanos en los que se destaque la capacidad de proveer servicios de alta calidad. Esta visión motiva la orientación del plan que proponemos, que aspira a conjugar práctica extensiva en habilidades directamente relacionadas con las necesidades que percibimos en el mercado laboral, con una sólida formación en los conceptos de base de la programación y con el énfasis en el cuidado de distintos criterios de calidad de los productos de software construidos.

Por lo mencionado esta propuesta se centra en formar Licenciados/as en Informática. Esta carrera además, otorga el título intermedio de Técnico/a Universitario/a en programación, que se obtiene a los dos años de cursada.

3. Objetivos de la carrera

La carrera tiene por objetivo la formación de profesionales capaces de concebir soluciones a un amplio espectro de problemas asociados a las tareas de análisis, diseño, programación e implantación de software. Con un fuerte y sólido conocimiento de la realidad nacional y social la carrera apunta a la formación de recursos humanos de excelencia capaces de desarrollar tecnología innovadora, respetando los factores legales, éticos, ambientales y de seguridad.

En función de lo anterior, los objetivos de la carrera son:

- Formar profesionales en el campo de la informática con un fundamento sólido en el desarrollo y calidad del software.
- Generar profesionales con capacidades de creación e innovación tecnológica.
- Formar recursos humanos con pensamiento crítico y conciencia social, con capacidades para vincularse con la industria del software regional, nacional e internacional.
- Generar una propuesta formativa que permita alcanzar capacidades de liderazgo y dirección de proyectos.
- Ofrecer al entorno territorial y regional una carrera que genere profesionales en el ámbito de la Informática, atendiendo a las problemáticas particulares que la caracterizan y a las necesidades de intervención al respecto.
- Garantizar una formación integral que desarrolle todas las dimensiones de la persona y habilite tanto para el desempeño social y laboral, como para el acceso a estudios de posgrado.
- Brindar una formación comprometida con los valores éticos y democráticos de participación, libertad, solidaridad, resolución pacífica de conflictos, respeto a los derechos humanos, responsabilidad, honestidad, valoración y preservación del patrimonio natural y cultural.

- Generar núcleos de investigación que promuevan la creación de conocimiento y la innovación científico-tecnológica en el ámbito de la disciplina informática.
- Generar proyectos de vinculación con el territorio que sean un servicio comunitario, un accionar concreto sobre las necesidades detectadas y un modo de acercar y profundizar los vínculos entre la Universidad y su entorno.

4. Perfil del egresado/a

4.1 Perfil del Técnico/a Universitario/a en Programación

El/la egresado/a es un técnico/a universitario/a cuya área de acción principal es la problemática de la construcción de software, que se corresponde con las tareas tradicionalmente conocidas como diseño y programación (codificación). El recorrido de la carrera le brinda experiencia en proyectos de desarrollo de aplicaciones organizacionales utilizando principalmente el paradigma de objetos. En este ámbito específico tiene los conocimientos para insertarse rápida y satisfactoriamente en el mundo laboral. De acuerdo al perfil propuesto, el egresado/a deberá:

- Tener una base conceptual sólida que le permita participar y desarrollar proyectos de software de distinta índole, tanto respecto del tipo como a las herramientas utilizadas; y también adaptarse a las nuevas herramientas que van apareciendo en el ámbito laboral.
- Comprender adecuadamente la pertinencia de construir software de acuerdo a varios parámetros de calidad, entre los que destacamos: claridad, inclusión de tests automáticos extensivos, extensibilidad, robustez frente a fallos, uso eficiente de recursos; también manejar los principales conceptos y herramientas requeridos para que sus productos cuenten con grados adecuados de calidad.
- Comprender y valorar la conveniencia de utilizar estándares abiertos y software libre en los entornos operativos y herramientas de desarrollo.
- Contar con conocimientos que le permitan asumir otras tareas además de la construcción (elucidación de requerimientos, despliegue, administración del entorno de ejecución) en diferentes tipos de proyectos.

4.2 Perfil del Licenciado/a en Informática

El licenciado/a en informática de la Universidad Nacional de Hurlingham está enfocado en dar respuestas a necesidades de la sociedad, empresas y organismos a través de procesos de puesta en funcionamiento de herramientas informáticas ya sea desde la construcción de éstas, como así también desde la adaptación de soluciones existentes, principalmente del ámbito del software libre. Posee una formación ética profesional, y una estrecha relación con el sector productivo con

capacidad de desarrollo de proyectos propios. Está preparado para intervenir en el ámbito público, privado y académico.

El/la licenciado/a en informática es un profesional universitario que:

- Tiene la capacidad de resolver problemas y liderar proyectos de desarrollo de software y aplicaciones de distinta índole y magnitud, como líder de equipos o como consultor.
- Dispone de saberes que le permiten intervenir en proyectos de redes de comunicaciones y seguridad informática.
- Posee conocimiento para asegurar la construcción de software de acuerdo a parámetros de calidad establecidos.
- Tiene la capacidad de promover, fomentar y valorar la aplicación de estándares abiertos y software libre en los entornos operativos y los procesos de desarrollo de software.
- Posee saberes relacionados con los procedimientos de pericias o auditorías informáticas.
- Adquiere capacidades para integrarse en equipos multidisciplinarios de desarrollo o investigación que integren soluciones de informática.
- Dispone de conocimientos sobre la gerencia de empresas cuyos principales procesos y plataformas que se sustenten en desarrollos informáticos de pequeña, mediana y gran escala.
- Tiene capacidad para liderar innovaciones tecnológicas que apliquen soluciones de supercómputo y/o modelos de inteligencia artificial con sustento en infraestructuras propias o en la nube.

5. Alcances

5.1. Alcances al título de Técnico/a Universitario/a en Programación

La Tecnicatura forma egresadas/os que puedan participar en el desarrollo de proyectos de software de cualquier porte y variadas características, adaptándose a distintos tipos de proyecto, formas de trabajo y herramientas. El grado de esta participación dependerá de las características de cada implementación.

Se habilita a un/a egresado/a para:

1. Diseñar, codificar, probar, medir y mantener programas de computadoras que forman parte de sistemas informáticos.
2. Llevar adelante todas las actividades de desarrollo e implantación de proyectos de software, incluyendo la elección de las herramientas a utilizar.
3. Integrar equipos interdisciplinarios que desarrollen procesos de análisis, diseño, despliegue y puesta en marcha de sistemas que integren tecnologías de la información.

5.2. Alcances de la Licenciatura en Informática

La Licenciatura forma profesionales con habilidades y conocimientos para:

- Proyectar, diseñar, dirigir, implementar y/o evaluar proyectos de sistemas de diferentes tipos y dominios:
 - Software de base: Sistemas Operativos, Sistemas Operativos Embebidos y/o Distribuidos.
 - Software de gestión de recursos: Sistemas de Bases de Datos, Procesamiento de Datos, Transformación y Transacción de grandes volúmenes de Datos. Gestión del Almacenamiento y la Seguridad de la Información.
 - Sistemas de comunicación de datos: redes de comunicaciones que interconectan sistemas de procesamiento de datos integrando software y hardware.
 - Sistemas de problemas generales: que aporten valor y soluciones mediante el procesamiento de datos a necesidades puntuales en diferentes ámbitos y aplicaciones.
- Realizar tareas de investigación científica básica y aplicada en Informática, participando como becario, Docente-Investigador o Investigador Científico/Tecnológico. Dirigir Proyectos, Laboratorios, Centros e Institutos de Investigación y Desarrollo en Informática.
- Efectuar las tareas de Auditorías de los Sistemas Informáticos. Realizar arbitrajes, peritajes y tasaciones relacionados con los Sistemas Informáticos.

5.3. Actividades profesionales reservadas al título de la Licenciatura en Informática

Las siguientes actividades profesionales reservadas al título son definidas por la Resolución N° 1254/2018.

1. Especificar, proyectar y desarrollar sistemas de información, sistemas de comunicación de datos y software cuya utilización pueda afectar la seguridad, salud, bienes o derechos.
2. Proyectar y dirigir lo referido a seguridad informática.
3. Establecer métricas y normas de calidad de software.
4. Certificar el funcionamiento, condición de uso o estado de lo mencionado anteriormente.
5. Dirigir y controlar la implementación, operación y mantenimiento de lo anteriormente mencionado.

5.4 Ejes transversales

Los ejes transversales son enunciados multidimensionales y transversales que requieren la articulación de conocimientos y de prácticas que fundamentan el ejercicio profesional. Estos enunciados son propuestos por la RM No 1546/2021 y se desarrollarán en los programas de las actividades curriculares, especificando el grado de profundidad en el tratamiento de los mismos. Cabe destacar que todos los ejes transversales son abordados en el transcurso de la carrera de manera gradual y progresiva.

- Identificación, formulación y resolución de problemas de informática.
- Concepción, diseño y desarrollo de proyectos de informática.
- Gestión, planificación, ejecución y control de proyectos de informática.
- Utilización de técnicas y herramientas de aplicación en la informática.
- Generación de desarrollos tecnológicos y/o innovaciones tecnológicas.
- Fundamentos para el desempeño en equipos de trabajo • Fundamentos para la comunicación efectiva.
- Fundamentos para la acción ética y responsable.
- Fundamentos para evaluar y actuar en relación con el impacto social de su actividad en el contexto global y local.
- Fundamentos para el aprendizaje continuo.
- Fundamentos para la acción emprendedora.

6. Condiciones de Ingreso

Los/as aspirantes a ingresar deberán:

- Poseer título de educación secundaria o equivalente expedido por escuelas de educación secundaria que cuenten con reconocimiento oficial. Excepcionalmente, podrán ingresar quienes tengan 25 (veinticinco) años o más (Art. 7 Ley de Educación Superior 24.521) y aprueben la evaluación establecida por la UNAHUR en la que se compruebe disponer de los conocimientos requeridos.
- Finalizar el Curso de Preparación Universitaria (CPU) que ofrece la Universidad.

7. Estructura curricular

7.1 Estructura por trayectos y campos formativos

La carrera está estructurada por trayectos y campos de formación. Estos refieren al modo en que se organizan y agrupan las unidades curriculares según sus propósitos formativos específicos. Componen la carrera cinco trayectos de formación que se complementan y articulan:

- Ciencias Básicas Generales y Específicas (CBGyE).
- Algoritmos y Lenguajes (AyL).
- Ingeniería de Software, Base de Datos y Sistemas de Información (ISBDSI).
- Arquitectura, Sistemas Operativos y Redes (ARSORE).
- Aspectos Sociales y Profesionales (ASyP).

A su vez, las actividades curriculares de acuerdo a las definiciones institucionales de la universidad, se organizan en campos de formación:

- CFC: campo de formación común.
- CFB: campo de formación básica.
- CFE: campo de formación específica.
- CIC: Campo de Integración curricular.

El **CFC** es común a todas las carreras de la UNAHUR y se compone de tres unidades curriculares que institucionalmente se considera que brindan los conocimientos y habilidades imprescindibles para el ejercicio de las profesiones. Se incluyen en el CFC las siguientes unidades curriculares:

- Cultura y alfabetización digital en la universidad
- Asignatura UNAHUR a elección entre las incluidas en la oferta definida anualmente por la Secretaría Académica.
- Inglés

Las unidades curriculares del CFC suman un total de 96 horas de interacción pedagógica y 179 horas de trabajo autónomo, que representan 275 horas totales y 11 créditos.

El **CFB** está conformado por 18 unidades curriculares obligatorias ,

El recorrido formativo de las materias del CFB plantea un abordaje profundo de contenidos fundamentales para poder abordar y conceptualizar desde el inicio mismo de la carrera principalmente los siguientes ejes transversales:

- Identificación, formulación y resolución de problemas de informática.
- Utilización de técnicas y herramientas de aplicación en la informática.
- Fundamentos para el desempeño en equipos de trabajo
- Fundamentos para la comunicación efectiva

- Fundamentos para evaluar y actuar en relación con el impacto social de su actividad en el contexto global y local
- Fundamentos para el aprendizaje continuo

A través del recorrido por las distintas unidades y actividades curriculares, se espera brindar a los y las estudiantes una formación teórica y práctica vinculada al contexto local, regional y global, comprometida socialmente y con una mirada política, crítica y reflexiva.

Se incluyen en el CFB las siguientes unidades curriculares:

- Introducción a lógica y problemas computacionales
- Matemática para informática 1
- Organización de las computadoras 1
- Tecnología y sociedad
- Programación estructurada
- Matemática para informática
- Lenguajes informáticos 1
- Organización de computadoras 2
- Estructuras de datos2
- Bases de datos
- Álgebra lineal
- Ejercicio Profesional en tecnología
- Matemática para Informática 3
- Redes de computadoras
- Sistemas y organizaciones
- Sistemas operativos
- Matemática para informática 4
- Probabilidad y estadística

Las unidades curriculares del CFB suman un total de 1248 horas de interacción pedagógica y 1402 horas de trabajo autónomo, que representan 2650 horas totales y 106 créditos.

El **CFE** es propio de la carrera y se compone de las unidades curriculares a las que refiere la especificidad de la titulación que se otorga.

Incluye saberes necesarios para la apropiación del conocimiento de la disciplina Informática. Incorpora la contextualización, la lógica y la legitimación de este conocimiento, así como los desarrollos científicos y técnicos propios; la articulación entre el campo específico y productivo, el contexto de desarrollo y su contribución al abordaje de problemáticas actuales.

Se incluyen en el CFE las siguientes unidades curriculares:

- Programación con objetos 1
- Programación con objetos 2

- Algoritmos
- Construcción de interfaces de usuario
- Estrategias de persistencia
- Ingeniería de software 1
- Desarrollo de aplicaciones
- Laboratorio de sistemas operativos y redes
- Lenguajes informáticos 2
- Arquitectura de software 1
- Ingeniería de software 2
- Lenguajes informáticos 3
- Seguridad de la información
- Computabilidad y complejidad
- Fundamentos de redes neuronales
- Lenguajes informáticos 4
- Formalización y generación de lenguajes
- Arquitectura de software 2
- Gestión de proyectos de desarrollo de software
- Aprendizaje automático
- Sistemas distribuidos y tiempo real
- Arquitectura de computadoras

Son 22 unidades curriculares con un total de 1504 horas de interacción pedagógica y 1646 de trabajo autónomo, que representan 3150 horas totales y 126 créditos.

El **CIC** se enfoca en la integración de saberes mediante la aplicación práctica de los mismos en proyectos de integración. En este campo se desarrollan actividades curriculares en las que las y los estudiantes se vinculan con temas y problemas específicos de sus profesiones.

Son parte de este campo las siguientes unidades curriculares:

- Práctica profesional supervisada
- Proyecto final

Las unidades curriculares del CIC suman un total de 160 horas de interacción pedagógica y 190 de trabajo autónomo, que representan 350 horas totales y 14 créditos.

Además, el plan de estudios incluye **Actividades Curriculares Acreditables (ACA)**, las cuales son un aporte a la flexibilidad. Son un conjunto de actividades consideradas valiosas para la formación, referidas al ámbito de la investigación, la extensión, la cultura, los eventos académicos, el deporte, el trabajo y de unidades curriculares electivas que se van acreditando con asignación parcial de créditos a medida que se cumplimentan. En tanto flexibles, no se determinan de antemano sino que se ofrecen a elección del estudiantado dentro del conjunto de posibilidades

que brinda y/o reconoce el Instituto de Tecnología e Ingeniería. Las ACA se regularán por medio de un reglamento específico.

Las ACA suman un total de 30 créditos (CRE), que se distribuyen de la siguiente manera:

- 10 créditos en unidades curriculares no incluidas en el plan de estudios.
- 10 créditos en experiencias formativas diversas.
- 10 créditos que se distribuirán según la definición del Instituto de tecnología e ingeniería.

Del total de créditos, el plan de estudios contempla que al menos el 10% (75 horas) de las ACA corresponde a interacción pedagógica, dependiendo de las actividades que desarrollen las y los estudiantes.

Se deben cumplimentar 12 CRE para obtener el título intermedio y 18 CRE más para obtener el título de grado.

El plan de estudios contempla un total de 3200 horas de interacción pedagógica y 287 créditos. Los trayectos formativos tienen en total 2752 horas de interacción pedagógica organizadas de la siguiente forma:

	Interacción Pedagógica
CBGyE	736
AyL	736
ISBDySO	512
ASOyR	640
ASyP	128

Las restantes 448 horas de interacción pedagógica corresponden a otros contenidos distribuidos en:

	Interacción pedagógica
Unidades curriculares del CFC	96
Actividades Curriculares Acreditables (ACA)	192
Prácticas Profesionales y Proyecto final	160

7.2 Estructura del plan de estudios

7.2.1. Estructura de la licenciatura en informática

D= Duración; C=Cuatrimestral; A=Anual; TF= Trayecto formativo; HIS=Horas de Interacción Semanal; HIT= Horas Interacción Totales; HITE= Horas Interacción Teórica; HIP= Horas Interacción Práctica; HTAT= Horas Trabajo Autónomo Totales; HT= Horas Totales; CRE= Créditos

Licenciado/a en Informática										
Cód.	Unidad curricular	TF	D	Carga Horaria						CRE
				HIS	HIT	HITE	HIP	HTAT	HT	
1° Año										
1	Introducción a lógica y Problemas Computacionales	CBGyE	C	4	64	32	32	111	175	7
2	Matemática para Informática 1	CBGyE	C	4	64	32	32	111	175	7
3	Organización de las Computadoras 1	ASOyR	C	4	64	32	32	111	175	7
4	Cultura y alfabetización digital en la universidad	OTRAS	C	2	32	16	16	68	100	4
5	Tecnología y Sociedad	ASyP	C	4	64	32	32	36	100	4
6	Inglés	OTRAS	C	2	32	16	16	68	100	4
7	Programación Estructurada	AyL	C	6	96	48	48	79	175	7
8	Matemática para Informática 2	CBGyE	C	4	64	32	32	111	175	7
9	Lenguajes Informáticos 1	AyL	C	4	64	32	32	111	175	7
10	Organización de Computadoras 2	ASOyR	C	4	64	32	32	61	125	5
2° Año										
11	Programación con Objetos 1	AyL	C	6	96	32	64	79	175	7
12	Estructuras de Datos	AyL	C	6	96	32	64	79	175	7
13	Bases de Datos	ISBDySO	C	4	64	30	34	111	175	7
14	Matemática para Informática 3	CBGyE	C	4	64	32	32	61	125	5
15	Redes de Computadoras	ASOyR	C	4	64	32	32	61	125	5
16	Programación con Objetos II	AyL	C	6	96	32	64	79	175	7
17	Asignatura UNAHUR	OTRAS	C	2	32	24	8	43	75	3
18	Sistemas y organizaciones	ISBDySO	C	4	64	32	32	86	150	6
19	Algoritmos	AyL	C	4	64	32	32	86	150	6
20	Sistemas Operativos	ASOyR	C	4	64	24	40	86	150	6
3° Año										
21	Construcción de Interfaces de Usuario	ISBDySO	C	4	64	30	34	111	175	7
22	Estrategias de Persistencia	ISBDySO	C	4	64	30	34	111	175	7
23	Ingeniería de Software 1	ISBDySO	C	4	64	32	32	111	175	7
24	Álgebra Lineal	CBGyE	C	4	64	32	32	61	125	5
25	Ejercicio Profesional en Tecnología	ASyP	C	4	64	32	32	36	100	4

26	Desarrollo de Aplicaciones	AyL	C	4	64	16	48	86	150	6
27	Laboratorio de Sistemas Operativos y Redes	ASOyR	C	4	64	24	40	86	150	6
28	Lenguajes Informáticos 2	AyL	C	6	64	32	32	61	125	5
29	Arquitectura de Software 1	ASOyR	C	4	64	32	32	86	150	6
30	Matemática para Informática 4	CBGyE	C	4	64	32	32	61	125	5
4° Año										
31	Ingeniería de Software 2	ISBDySO	C	4	64	32	32	61	125	5
32	Probabilidad y Estadística	CBGyE	C	6	96	48	48	29	125	5
33	Lenguajes Informáticos 3	AyL	C	4	64	32	32	61	125	5
34	Seguridad de la Información	ASOyR	C	4	64	32	32	61	125	5
35	Computabilidad y Complejidad	CBGyE	C	4	64	32	32	61	125	5
36	Fundamentos de Redes Neuronales	CBGyE	C	4	64	32	32	61	125	5
37	Lenguajes Informáticos 4	AyL	C	6	96	48	48	54	150	6
38	Formalización de Lenguajes y Generación de Código	CBGyE	C	4	64	32	32	61	125	5
39	Arquitectura de Software 2	ASOyR	C	4	64	24	40	61	125	5
40	Práctica Profesional Supervisada	OTRAS	C	4	64	16	48	111	175	7
5° Año										
41	Gestión de Proyectos de Desarrollo de Software	ISBDySO	C	4	64	32	32	61	125	5
42	Aprendizaje Automático	CBGyE	C	4	64	32	32	86	150	6
43	Sistemas Distribuidos y Tiempo Real	ASOyR	C	4	64	32	32	61	125	5
44	Arquitectura de Computadoras	ASOyR	C	4	64	32	32	61	125	5
45	Proyecto final de Licenciatura	OTRAS	C	6	96	32	64	79	175	7
Actividades Curriculares Acreditables (ACA)					192	'	'	558	750	30
					3200	1386	1622	3975	7175	287
TITULACIÓN FINAL: Licenciado/a en Informática										

Las correlatividades serán aprobadas por el órgano de gobierno correspondiente en norma ad-hoc.

7.2.2. Estructura del título del intermedio

D= Duración; C=Cuatrimstral; A=Anual; TF= Trayecto formativo; HIS=Horas de Interacción Semanal; HIT= Horas Interacción Totales; HITE= Horas Interacción Teórica; HIP= Horas Interacción Práctica; HTAT= Horas Trabajo Autónomo Totales; HT= Horas Totales; CRE= Créditos

Técnico/a Universitario/a en programación										
Cód.	Unidad curricular	TF	D	Carga Horaria						CRE
				HIS	HIT	HITE	HIP	HTAT	HT	
1° Año										
PRIMER CUATRIMESTRE										
1	Introducción a lógica y Problemas Computacionales	CBGyE	C	4	64	32	32	111	175	7
2	Matemática para Informática 1	CBGyE	C	4	64	32	32	111	175	7
3	Organización de las Computadoras 1	ASOyR	C	4	64	32	32	111	175	7
4	Cultura y alfabetización digital en la universidad	OTRAS	C	2	32	16	16	68	100	4
SEGUNDO CUATRIMESTRE										
6	Inglés	OTRAS	C	2	32	16	16	68	100	4
7	Programación Estructurada	AyL	C	6	96	48	48	79	175	7
8	Matemática para Informática 2	CBGyE	C	4	64	32	32	111	175	7
9	Lenguajes Informáticos 1	AyL	C	4	64	32	32	111	175	7
2° Año										
PRIMER CUATRIMESTRE										
11	Programación con Objetos 1	AyL	C	6	96	32	64	79	175	7
12	Estructuras de Datos	AyL	C	6	96	32	64	79	175	7
13	Bases de Datos	ISBDySO	C	4	64	30	34	111	175	7
SEGUNDO CUATRIMESTRE										
16	Programación con Objetos II	AyL	C	6	96	32	64	79	175	7
17	Asignatura UNAHUR	OTRAS	C	2	32	24	8	43	75	3
18	Sistemas y organizaciones	ISBDySO	C	4	64	32	32	86	150	6
3° Año										
PRIMER CUATRIMESTRE										
21	Construcción de Interfaces de Usuario	ISBDySO	C	4	64	30	34	111	175	7
22	Estrategias de Persistencia	ISBDySO	C	4	64	30	34	111	175	7
23	Ingeniería de Software 1	ISBDySO	C	4	64	32	32	111	175	7
	Actividades Curriculares Acreditables (ACA)				64	-	-	236	300	12
					1184	514	606	1705	3000	120
TITULACIÓN INTERMEDIA: Técnico/a Universitario/a en Programación										

Las correlatividades serán aprobadas por el órgano de gobierno correspondiente en norma ad-hoc.

8. Seguimiento curricular

El seguimiento curricular es un proceso continuo de monitoreo y evaluación para asegurar que el plan de estudios cumpla con los cometidos para los que fue pensado.

El objetivo es contar con un insumo que permita identificar aspectos a mejorar y tomar decisiones para optimizar los procesos de enseñanza y de aprendizaje.

La Comisión de Carrera en conjunto con la Secretaría Académica será la encargada de realizar el seguimiento curricular.

9. Formación Práctica

La formación práctica es concebida en este plan de estudios, de manera tal de permitir que el estudiante a lo largo de su carrera, incorpore saberes teóricos y prácticos, que les permitan desarrollar competencias profesionales para un adecuado desempeño en relación a las actividades reservadas al título y a los alcances definidos para el perfil de egresado de la carrera.

Los criterios que rigen la intensidad de la formación práctica son:

- Gradualidad y complejidad. El aprendizaje constituye un proceso de reestructuraciones continuas, que posibilita de manera progresiva alcanzar niveles cada vez más complejos de comprensión e interpretación de la realidad.
- Integración de teoría y práctica. La intervención en la problemática específica de la Informática contempla ámbitos o modalidades curriculares de articulación e integración teórico-práctica que, además de recuperar el aporte de diferentes disciplinas, propicien la permanente reflexión sobre la práctica en situaciones concretas que requieren el desarrollo de soluciones informáticas a problemas del mundo real.
- Resolución de situaciones problemáticas. El proceso de apropiación del conocimiento científico o tecnológico requiere el desarrollo de la capacidad de identificar y resolver problemas del mundo real que requieren de la Informática, dentro de un enfoque sistémico e interdisciplinario.

En este sentido la intensidad de la formación práctica garantiza que el estudiante logre introducirse a los estudios universitarios en Informática, interpretar los problemas del mundo real relacionados con la aplicación de la disciplina e intervenir de manera efectiva para resolver los mismos.

La formación práctica se desarrollará en diferentes dimensiones. Por un lado, facilitando que el estudiante se familiarice con la Universidad, la organización y funcionamiento de las instituciones de enseñanza de la Informática y su vinculación

con la realidad. Asimismo, en esta dimensión se desarrollan habilidades prácticas en actividades experimentales y de resolución de problemas que acercan la realidad de las organizaciones de Software y Servicios Informáticos (empresas privadas / organismos públicos).

Por otro lado, se promueve la interpretación de la realidad vinculada con el profesional informático a través del diagnóstico y análisis de problemas, articulando la teoría con la práctica. Por último, la intervención crítica se promueve a partir de prácticas formativas contextualizadas. Estas prácticas incluyen la participación del estudiante en actividades de carácter científico, tecnológico y/o experiencias de intervención profesional, que permitan resolver problemas del sector SSI, en el contexto del perfil del graduado definido institucionalmente y en el marco de las actividades reservadas al título.

En este plan de estudios de las 3200 horas previstas por la resolución 1546/21 del ME, hay un total de 1622 horas destinadas a la formación práctica distribuidas de la siguiente manera:

CBGyE	Ciencias Básicas Generales y Específicas	368
AyL	Algoritmos y Lenguajes	416
ISBDySO	Ingeniería de Software, Base de Datos y Sistemas de Información	278
ASOyR	Arquitectura, Sistemas Operativos y Redes	344
ASyP	Aspectos Sociales y Profesionales	64
OTRAS		152

La formación práctica se consolida e integra mediante las actividades curriculares del CIC:

- La Práctica Profesional Supervisada (PPS) es una actividad formativa en la cual el alumno realiza una incorporación supervisada y gradual al trabajo profesional, a través de su inserción a una realidad o ambiente laboral específico relacionado con la informática y de esta manera aplica integralmente los conocimientos adquiridos a lo largo de su formación académica. El Reglamento de Práctica Supervisada de la Universidad de Hurlingham regula los objetivos, metodología, acciones, plan de trabajo, actividades, evaluación, docente responsable y lugar de realización. La

supervisión la realiza un tutor docente y deberá acreditarse un tiempo mínimo de horas de práctica profesional dentro de la misma universidad, en instituciones, organismos, sectores productivos, y/o de servicios.

- El Proyecto Final consta de la realización de un trabajo técnico y/o científico y/o desarrollo tecnológico y/o aquel trabajo de carácter analítico - científico, de elaboración y conclusiones personales relacionado con las incumbencias profesionales e integrador de los conocimientos adquiridos, que debe realizar y presentar todo alumno/a para obtener el grado de Licenciado. El reglamento de Proyectos integradores de la Universidad de Hurlingham regula los objetivos, características, requisitos previos, elección del tema, dirección, responsable de la asignatura, desarrollo del proyecto, finalización y examen. El Proyecto será guiado y supervisado por un docente tutor.

10. Contenidos mínimos

El plan de estudios se orienta por los ejes transversales fomentando el desarrollo de habilidades blandas y contenidos que permitan el entender las variables económicas del sector en el ámbito regional, nacional y global así como la potencialidad de desarrollar capacidades de gerenciamiento.

Así mismo se estimulan la formación y relativa al impacto social y ecológico, así como habilidades que estimulen la capacidad de análisis, de síntesis y el espíritu crítico del estudiante, que despierten su vocación creativa y entrenen para el trabajo en equipo y la valoración de alternativas.

Cultura y alfabetización digital en la universidad:

Derechos y ciudadanía digital. Reflexión crítica sobre la cultura contemporánea. Entornos y plataformas digitales de aprendizaje. Herramientas de colaboración en ambientes digitales. Recursos de información en la era digital: búsquedas efectivas y evaluación crítica de fuentes. Producción, uso y distribución de contenidos digitales académicos. Exploración y apropiación de tendencias y tecnologías emergentes.

Asignatura UNAHUR:

Las asignaturas UNAHUR son propuestas flexibles que realiza la Universidad atendiendo a coyunturas y propósitos particulares y los/las estudiantes pueden elegir entre las incluidas en la oferta definida anualmente por la Secretaría Académica. En el Anexo 1 se detallan los contenidos mínimos de algunos ejemplos de asignaturas UNAHUR actualmente ofrecidas.

Inglés:

"Introducción a la lectura de textos auténticos de géneros específicos de las distintas disciplinas. Palabras conceptuales y estructurales. Estrategias y técnicas de lectura. El género textual. El sintagma nominal. Morfología: sufijos y prefijos. Uso del diccionario y otro material de referencia. Organización textual, tema y despliegue temático. Elaboración del tópico de un texto. La definición. Cohesión gramatical y lexical: referencia, conjunciones y repetición. Las instrucciones. La descripción. La narración. Las oraciones condicionales. El sintagma verbal: tiempo, voz y aspecto. Formas no conjugadas. Tiempos verbales simples, progresivos y perfectivos. Modales simples y perfectivos. La voz pasiva."

Introducción a lógica y problemas computacionales:

Qué es la informática: hardware vs. software, historia de las computadoras, presente, posibles escenarios futuros. Historia del software y los lenguajes de programación: qué son los paradigmas de programación: imperativo, orientado a objeto y funcional.

Qué es un programa. Entornos de desarrollo y ejecución. Principios de la programación imperativa: comandos (acciones), estructuras de control de flujo de programas (secuencia, repetición simple, repetición condicional, alternativa condicional en comandos). Sensores booleanos. Conectivas booleanas. Sensores numéricos. División en subtareas como metodología para la resolución de problemas complejos, y necesidad de dar estructura a un programa no trivial.

Matemática para informática 1:

Elementos de lógica proposicional y de primer orden: Enfoque sintáctico y semántico. Técnicas de prueba. Teoría de las Estructuras Discretas. Cuantificadores. Condicionales Asociados. Razonamientos Deductivos. Leyes de Inferencia. Teoría básica de conjuntos. Problemas de Conteo. Números enteros: Teoría de número.

Organización de las computadoras 1:

Historia de la computación. Definición de computadora. La "información" en el mundo real (magnitudes analógicas y cantidades discretas) y su representación como "datos" binarios dentro del computador. Sistemas de representación numérica (SRN) decimales y binarios. Generalización a SRN posicionales en otras bases (octal, hexa o genérico "r"). Representación de números enteros (con signo) y racionales en sistemas de punto fijo y punto flotante. Errores al representar números racionales e irracionales en un sistema de ancho finito. Sistemas de representación

alfanumérica. Representación de variables lógicas. Álgebra binaria: suma, resta y conceptos de multiplicación. Lógica digital: Axiomas y propiedades del álgebra de Boole, operaciones y compuertas lógicas. Circuitos combinacionales genéricos. Equivalencia entre la tabla de verdad y las funciones canónicas. Semisumadores y sumadores. Flags. Introducción conceptual a los módulos funcionales en los que se organiza una computadora: ALU, registros, bancos de registros, memoria y dispositivos de entrada/salida. Los caminos de datos (buses de datos, direcciones y control). Conceptos de lenguaje de máquina y ensamblador. Programas ensambladores. Relación entre lenguajes de alto nivel y código de máquina. Programas compiladores e intérpretes. Ejemplos de arquitecturas reales. Diferencias entre microprocesadores, microcontroladores y sistemas embebidos.

Tecnología y sociedad:

Computación y Sociedad. Modelos de desarrollo científico y tecnológico. El proceso de producción de conocimiento. Etapas históricas. Historia de la computación. Ciencia, tecnología y economía. Sus interrelaciones. Etapas del desarrollo científico y tecnológico. El papel de la Universidad. La investigación científica y tecnológica en la actualidad. Software libre y sociedad. El posicionamiento del software libre y el código abierto. Cuestiones éticas y sociales. Sesgos algorítmicos. Impacto del uso de la IA. Definiciones de principios éticos. Regulaciones nacionales e internacionales. Posicionamiento de los/las profesionales en la resolución de problemas mediante IA.

Programación estructurada:

Qué es un programa. Entornos de desarrollo y ejecución. Principios de la programación imperativa: Comandos (acciones) y Expresiones (valores), estructuras de control de flujo de programas (secuencia, repetición simple, repetición condicional, alternativa condicional en comandos y expresiones), estado, tipos de datos (números, booleanos, cadenas, enumerativos simples, con estructura mediante campos). División en subtarear como metodología para la resolución de problemas complejos, y necesidad de dar estructura a un programa no trivial. Generalización de programas mediante parametrización. Resolución de problemas y algoritmos mediante programas. Precondiciones como metodología para el desarrollo de software robusto. Buenas prácticas de programación (indentación, documentación, elección de nombres).

Matemática para informática 2:

Números naturales: Inducción matemática. Relaciones binarias. Estudio de las propiedades de una relación binaria. Clasificación. Relaciones de orden. Relaciones

de equivalencia. Principios básicos de recuento: Regla del producto y regla de la suma. Factorial de un número. Propiedades. Análisis Combinatorio simple y con repetición.

Lenguajes informáticos 1:

Introducción a los lenguajes de marcado, su estructura y sintaxis básica. Organización y estructuración de documentos para la creación de esquemas completos de sitios web. Separación del contenido y su presentación mediante lenguajes de definición de estilos, aplicando principios de diseño y accesibilidad. Introducción a lenguajes de programación del lado del cliente para la incorporación de funcionalidades dinámicas en sitios web, incluyendo la manipulación de documentos estructurados, la gestión de eventos y la validación de datos.

Organización de computadoras 2:

Combinacionales genéricos. Circuitos combinacionales específicos: Sumador, decodificador, multiplexor, demultiplexor, detector de paridad, comparador de magnitud y codificador de prioridades. Biestables. Circuitos secuenciales. Celda de memoria, registro de desplazamiento y contador. Organización y arquitectura del computador. Unidades funcionales. Unidad aritmético lógica (ALU). Memoria y sus niveles de jerarquía. Subsistemas de entrada y salida. Unidad de control y camino de datos.

Arquitectura. ISA. Instrucciones en código de máquina. Von Neumann. Harvard. RISC. CISC. Ciclos de instrucción. Tipos de direccionamiento. Lenguaje de transferencia de registros(RTL). Lenguaje de máquina. Lenguaje ensamblador. Código fuente y código objeto. Ensambladores, intérpretes y compiladores. Conceptos de arquitecturas superescalares y multiprocesamiento."

Programación con objetos 1:

Conceptos fundantes del paradigma: objeto y mensaje. Concepto de polimorfismo en objetos, comprensión de las ventajas de aprovecharlo. Protocolo/interfaz, concepto de tipo en objetos, comprensión de que un objeto puede asumir distintos tipos. La interfaz como contrato al que se compromete el objeto modelado, posibilidad de reforzar ese contrato. Estado en el paradigma de objetos: referencias, conocimiento, estado interno. Métodos, clases, herencia simple, method lookup. Conceptos de responsabilidad y delegación. Colecciones: conceptualización como objetos, caracterización a partir de los conceptos de protocolo y responsabilidad, protocolo, acceso a sus elementos. Testeo automático y repetible. Nociones básicas

sobre manejo de errores. Interrupción del flujo de ejecución: modelado mediante estructuras de control, concepto de excepción.

Estructuras de datos:

Programación estructurada y modular. Programas recursivos. Representación de datos en memoria. Paso de variables por valor y por referencia. Tipos abstractos de datos. Estructuras de datos. Estructuras contenedoras: Vectores, matrices, pilas, colas, lista, diccionarios, árboles y grafos. Implementación de estructuras de datos estáticas. Uso dinámico de memoria. Listas y árboles implementados con punteros. Análisis, eficiencia e implementación de estructuras de datos. Algoritmos de recorrido, búsqueda y ordenamiento. Nociones básicas de algoritmos sobre grafos.

Bases de datos:

Qué es un modelo de datos, modelos conceptuales, lógicos y físicos. Modelo de entidad-relación: conceptos básicos. Modelo relacional: tabla, atributo, dominio, valor, fila; restricciones de integridad; operaciones que se pueden hacer. SQL: concepto de lenguaje de consulta, sintaxis, concepto de join, agrupamientos, subqueries, joins parciales. Sistemas de Bases de Datos. Diseño y administración de Sistemas de Bases de Datos. Escalabilidad, eficiencia y efectividad. Lenguajes de DBMS. Transacción: concepto, demarcación de transacciones.

Matemática para Informática 3:

Funciones. Representación gráfica. Dominio e Imagen. Distintos tipos de Funciones. Función inversa. Composición de funciones. Límite. Continuidad. Derivada. Aplicaciones de la Derivada. Estudio completo de funciones Diferencial. Teoremas del Cálculo Diferencial. Regla de L'Hôpital. Problemas de Optimización. Integral Indefinida. Integral Definida. Cálculo de Áreas.

Redes de computadoras:

Concepto de red de computadoras, redes y comunicación. Modelos OSI y TCP/IP. Conceptos de protocolo y de servicio. Nivel físico: repetidor y hub, cableado estructurado. Nivel de enlace: concepto de medios físicos, tramas, bridge y switch, enlaces inalámbricos, vlans, spanning tree. Nivel de red: concepto de ruteo, topologías, ruteo estático, algoritmos y protocolos de ruteo dinámico, protocolo IPv4, resolución de direcciones. Nivel de transporte: funciones, protocolos UDP y TCP, multiplexación, concepto de socket, control de congestión, servicios de capa de aplicación (http, dhcp, dns, smtp, etc.). El modelo computacional de la Web.

Estándares utilizados en Internet, concepto de RFC. Concepto e implementación de las VPN. Administración de redes: servicios, firewalls. Sistemas cliente/servidor y sus variantes.

Sistemas y organizaciones:

Fundamentos de la Teoría General de Sistemas. Enfoque Sistémico. Modelos Conceptuales. Las organizaciones. Análisis de las organizaciones. La Organización como sistema. Los Sistemas de Información. Procesos de Negocio. Modelos de negocio. Sistemas de Información asociados a los Procesos de Negocio.

Sistemas operativos:

Introducción a los sistemas operativos: función de abstracción del hardware; organización, estructura y servicios de los SO. Sistemas operativos: de tiempo real, embebidos (embedded), distribuidos. Sistemas batch / Multiprogramación / / Sistemas paralelos. Conceptos de proceso, thread y planificación. Concurrencia de ejecución. Interbloqueos. Comunicación y cooperación entre procesos. Deadlocks. Planificación: Algoritmos, criterios. Multiprocesamiento. Administración de memoria: Espacio lógico y físico, swapping, alocaión contigua, paginación, segmentación. Memoria virtual: Paginación bajo demanda, algoritmos de reemplazo de página, thrashing. Sistemas de archivos: Sistemas de archivos. Protección. Manejo de directorios. Concepto de Proceso. Planificación de Procesos. Protección: objetivos, dominio de protección, matriz de acceso y sus implementaciones. Prácticas con distintos sistemas operativos.

Programación con objetos 2:

Aproximación al diseño de software, entendiendo el diseño como un proceso de toma de decisiones críticas. Noción de decisión de diseño y su impacto en el desarrollo de software. Conceptos de acoplamiento y cohesión, y los problemas que surgen de un grado de acoplamiento inadecuado. Vinculación entre las ideas básicas de diseño y el paradigma de objetos. Características deseadas en un diseño de objetos y cómo lograrlas. Introducción y aplicación de patrones de diseño. Nociones sobre el proceso de diseño, incluyendo el manejo de eventos y metaprogramación. Uso de un entorno integrado de desarrollo de software (IDE). Notación UML para diagramas de clases, objetos y secuencias. Implementación de testeo unitario y automático. Manejo de errores y su impacto en el diseño de software.

Algoritmos:

Noción de algoritmo, ejemplos de algoritmos (criba de Eratóstenes, mcd, etc). Criterios de selección de un algoritmo. Notación O y W. Análisis teórico del tiempo de ejecución de un algoritmo Análisis práctico del tiempo de ejecución de un algoritmo. Algoritmos Divide y Vencerás. Recursividad. Análisis de procedimientos recursivos. Algoritmos Basados en Programación Dinámica. Estrategias de diseño de algoritmos. Algoritmos Greedy. Algoritmos de Precondicionamiento y Transformación del Dominio. Algoritmos de programación matemática. Uso de Heurísticas en Algoritmos. Algoritmos numéricos y propagación del error. Casos: algoritmo de Huffman, encriptación, compresión, búsqueda, actualización, ordenamiento, estructuras de datos y algoritmos, árboles estrella, matrices. Algoritmos sobre grafos (DFS, BFD, Prim, Kruskal, Dijkstra, Floyd, sort topológico, etc). Algoritmos básicos sobre cadenas: matching, alineamiento, sufijos. Algoritmos concurrentes, distribuidos y paralelos. Estrategias de implementación.

Construcción de interfaces de usuario:

Variantes en arquitecturas de sistema respecto de la interfaz de usuario (IU): aplicación centralizada, cliente-servidor o distribuida; ejecución en un cliente de aplicación (browser, flash, otros) o mediante un programa específico; concepto de RIA. Arquitecturas web, protocolos y tecnologías asociadas. Modelos de interacción de la IU con su entorno: interfaces orientadas a eventos, pedido-respuesta, basadas en continuations. Aplicaciones client-initiative y application-initiative. Componentes gráficos usuales en interfaces de usuario. Vinculación entre la IU y el modelo de dominio subyacente. Problemática asociada a transformaciones, validaciones, manejo de errores, excepciones, transacciones e identidad. Impacto de la distribución de aplicaciones en la IU, comunicación sincrónica y asincrónica. Navegación y manejo del estado conversacional. REST, estado en sesión. Nociones de usabilidad. Diseño centrado en el usuario.

Estrategias de persistencia:

Problemas vinculados al acceso concurrente a una base de datos (BD). Performance en el acceso a una BD. Modelado y calidad de datos. Concepto de índice. Conceptos de usuario y permiso en una base de datos. Bases de objetos: concepto, panorama, experimentación práctica, comparación con bases de datos relacionales. Base de datos distribuidas para grandes volúmenes de datos, acceso a datos. Transacciones distribuidas. Interacción entre un programa y un mecanismo de persistencia: nociones básicas, problemáticas generales. Mecanismos de acceso y recuperación de objetos persistidos en bases de datos relacionales. Actualización del estado persistente: reachability, cascada. ORM y problemas de mapeo: herencia,

relaciones n-m, estrategias no estándar. Transacciones a nivel aplicación y de negocio, concepto de unit of work. Cuestiones de performance y concurrencia al acceder a un mecanismo de persistencia desde un programa, lazyness, cache, versionado, bloqueo optimista y pesimista.

Ingeniería de software 1:

Ingeniería de software. Software vs. Sistemas de Información. El proceso del software. modelos de procesos o ciclos de vida para el desarrollo de software. Metodologías ágiles: actividades, productos, formas de articulación, roles. Ejemplo: Scrum. Metodologías estructuradas: actividades, productos, formas de articulación, roles. Ejemplo: UP. Similitudes y diferencias entre metodologías ágiles y estructuradas. Introducción a los requerimientos. Requerimientos funcionales vs. Requerimientos no funcionales. Herramientas de modelado.

Álgebra lineal:

Matrices y sus operaciones. Inversa de una matriz cuadrada. Sistemas de Ecuaciones Lineales. Función Determinante. Métodos de solución de un sistema de ecuaciones lineales: Gauss-Jordan y Cramer. Sistemas de ecuaciones lineales homogéneos. Espacios vectoriales. Subespacios vectoriales. Combinación lineal y espacios generados. Independencia lineal. Base y dimensión de un espacio vectorial. Espacios vectoriales genéricos. Cambio de bases. Transformaciones lineales. Matriz de una transformación lineal. Autovalores y autovectores. Diagonalización de matrices.

Ejercicio Profesional en tecnología:

Aspectos Legales y Sociales, Consideraciones generales sobre el ordenamiento jurídico. Responsabilidad ética y profesional. Elementos de las relaciones económicas jurídicas. La propiedad intelectual. Conceptos generales de contratos. Licenciamiento de software y contratos informáticos. Documento digital, firma digital y derecho de Internet. Régimen legal de las bases de datos. Responsabilidad penal: delitos informáticos. Actuación judicial del licenciado en informática. Nociones de auditoría y peritaje. Gobernanza IT. Seguridad laboral en entornos tecnológicos. Impacto ambiental de las actividades en Informática.

Desarrollo de aplicaciones:

Validación y testing de software. Tests de integración. Problemática específica para la automatización de tests de integración, persistencia, interfaz de usuario. Técnicas

para diagnóstico de problemas: stacktraces, breakpoints, watchpoints. Manejo de excepciones. Relación con unittesting. Reingeniería de software. Técnicas de refactorización sobre un proyecto funcionando. Migraciones y actualizaciones. Versionado y compartición de programas fuente. Repositorios de código centralizados y distribuidos. Versionado y compartición de bibliotecas y ejecutables. Administración de entregables y dependencias. Repositorios de bibliotecas. Integración continua. Control de cambios. Trazabilidad de requerimientos, errores y cambios de funcionalidad. Herramientas para la administración integral de cambios y correcciones. Estrategias de implementación de sistemas de software. Aplicación en un proyecto mediano de desarrollo de software.

Laboratorio de sistemas operativos y redes:

Instalación, configuración y operación de distintos servicios relacionados con Internet: servidores de aplicaciones, servidor y cliente de mail, servidor y cliente FTP, firewalls, etc. Servicios de directorio, servidores LDAP, uso desde aplicaciones. Gestión de usuarios y control de accesos en un entorno operativo, impacto en la instalación de aplicaciones, posibilidad de compartir recursos. Sistemas de backup automatizados, políticas de criticidad. Instalación, configuración y operación de repositorios de código. Monitoreo de redes, protocolo SNMP. Técnicas de transmisión de datos, modelos, topologías, algoritmos de ruteo y protocolos. Sistemas operativos de redes. Computación orientada a redes. Sistemas colaborativos.

Lenguajes informáticos 2:

Estudio de los principales paradigmas de programación, sus modelos de cómputo y estructuras de datos asociadas. Análisis de lenguajes de programación en función de sus características y propósitos. Introducción a los sistemas de tipos y mecanismos de chequeo de tipos, esquemas de binding y variantes del paradigma orientado a objetos, incluyendo técnicas de metaprogramación, reflexión y programación orientada a aspectos. Estudio de la programación funcional: funciones como valores, polimorfismo, currificación, recursión estructural, e inferencia de tipos. Introducción a la construcción y clasificación de lenguajes específicos de dominio, su integración en aplicaciones multilenguaje y multiparadigma, y el diseño de programas basados en especificaciones mediante técnicas de transformación.

Arquitectura de software 1:

Arquitecturas en diferentes metodologías de desarrollo. Requerimientos funcionales y no funcionales, restricciones, influencias, entorno social y técnico, estándares,

herramientas disponibles. Objetivos de una arquitectura. Estilos arquitectónicos. Arquitectura de dominio. Arquitectura y Diseño. Patrones. Patrones arquitecturales para la interfaz de usuario. Integración con el dominio. Internacionalización. Arquitecturas orientadas a servicios. Arquitecturas extensibles. Arquitecturas basadas en plugins. Arquitecturas de seguridad. Estrategias de verificación de arquitecturas. Arquitecturas concurrentes y distribuidas. Herramientas tecnológicas para soportar las decisiones arquitectónicas.

Matemática para informática 4:

Funciones de varias variables. Composición de funciones de varias variables. Curvas y superficies parametrizadas. Límite, continuidad y derivabilidad. Diferenciabilidad. Derivación de funciones implícitas. Extremos. Ecuaciones diferenciales ordinarias. integral doble.

Ingeniería de software 2:

Ingeniería de requerimientos. Estrategias para la extracción de requerimientos. Herramientas conceptuales para la organización de requerimientos en modelos. Análisis basado en casos de uso. Análisis orientado a objetos. Representación avanzada de los requerimientos mediante el uso del lenguaje natural. Estrategias de análisis en metodologías ágiles. Requerimientos funcionales vs. Requerimientos no funcionales. Concepto. Características. Tratamiento. Calidad del software. Métricas: qué son, qué miden, para qué y cuándo sirven. Estimación de esfuerzos. Validación de requerimientos, relación con testing. IA aplicada a la Ingeniería de Requerimientos.

Probabilidad y estadística:

Probabilidad y estadística. Estadística descriptiva. Modelos determinísticos y estocásticos. Distribución de probabilidades sobre un espacio muestral. Variables aleatorias discretas y continuas. Distintos tipos de distribuciones. Inferencia estadística. Intervalos de confianza. Varianza. Regresión lineal. Coeficientes de correlación. Ensayos de hipótesis. Números aleatorios. Método Montecarlo.

Lenguajes informáticos 3:

Estudio de los fundamentos de la programación concurrente, el paralelismo y la programación basada en eventos. Modelos de concurrencia y sus motivaciones; diferencias conceptuales y operativas entre concurrencia, paralelismo y eventos. Modelos de memoria compartida y pasaje de mensajes, incluyendo mecanismos de

sincronización, exclusión mutua y comunicación entre procesos e hilos. Análisis de problemas asociados a la concurrencia: condiciones de carrera, inanición, interbloqueos, seguridad y progreso. Exploración de modelos de interacción como cliente-servidor y productor-consumidor. Programación orientada a eventos en entornos concurrentes. Aplicación de técnicas de concurrencia, paralelismo y programación basada en eventos en lenguajes de programación contemporáneos. Introducción a la programación sobre unidades de procesamiento gráfico (GPU) y estudio de criterios de eficiencia energética en entornos de ejecución concurrente y paralelo.

Seguridad de la información:

Introducción a la Seguridad de la Información. Conceptos fundamentales y objetivos. Gestión de la Seguridad de la Información. Riesgo: análisis y tratamiento. Seguridad en Redes, elementos de criptografía. Criptografía Simétrica y Asimétrica. Algoritmos de Hash. Infraestructura de Clave Pública. Certificados digitales. Seguridad en Redes. Objetivos. Ataques, Servicios y Mecanismos de Seguridad. Seguridad en Redes Inalámbricas. Control de Acceso Lógico. Controles físicos de seguridad: seguridad en el centro de cómputos. Seguridad en las operaciones. Gestión de usuarios. Control de cambios. Métodos de Evaluación de seguridad: Auditorías, Evaluaciones funcionales, Vulnerability Assessment y Penetration Test. Gestión de Incidentes. Seguridad en Aplicaciones. Vulnerabilidades. Software malicioso. Problemática de las aplicaciones WEB. Leyes, regulaciones y estándares. Marcos legales nacional e internacional. Privacidad, Integridad y seguridad en sistemas de información.

Computabilidad y complejidad:

Máquinas de Turing. Máquinas Algorítmicas. Conceptos Básicos de Teoría de Computabilidad y Complejidad: Problemas computables y no computables. Problema de la parada. Problemas tratables e intratables. Conjuntos decidibles, Funciones recursivas. Conjuntos recursivamente enumerables. Reducciones many-one. Clases L, P, PSPACE, NP, NP - Completitud. Análisis de Algoritmos: Análisis asintótico, comportamiento en el mejor caso, caso promedio y peor caso. Notación $O()$. Balance entre tiempo y espacio en los algoritmos. Análisis de Complejidad de Algoritmos. Teoría de base de datos.

Fundamentos de redes neuronales:

Introducción. Origen, historia y fundamentos biológicos. Funcionamiento de las redes neuronales artificiales. Problemas que pueden resolver. Modelos. Componentes. Funciones de activación. Red neuronal artificial (RNA). Arquitectura

de las RNA. Aprendizaje. Perceptrón simple y multicapa. Descenso de gradiente. Algoritmos de propagación.

Lenguajes informáticos 4:

Qué es la inteligencia artificial (IA), historia, evolución. Definiciones y enfoques y propósitos. Paradigmas de la IA: conexionista, clásico. Problemas de IA. Dominios: técnicos, formales, cognitivos. Fundamentos de inteligencia artificial simbólica y no simbólica. Lógica Proposicional: Lenguaje, Semántica, Mecanismo Deductivo, Metateoremas, Lógica trivaluada. Lógica de Primer Orden: Lenguaje, Semántica, Sistema axiomático, Metateoremas, Indecidibilidad. Programación lógica: Resolución en lógica de primer orden, PROLOG. Especificación de Programas: Especificación e implementación de programas, Lógica de Hoare, Corrección de programas. Verificación de algoritmos.

Formalización y generación de lenguajes:

Lenguajes y gramáticas. Gramáticas e isomorfismos. Jerarquía de Chomsky. Lenguajes regulares. Autómatas. Expresiones regulares. Minimización de autómatas. Analizadores lexicográficos. Lenguajes independientes de contexto. Árboles de derivación. Autómatas de pila. Lenguajes determinísticos. Lenguajes tipo 1 y tipo 0. Máquinas asociadas. Máquinas de Turing. Problema de la detención. Estructura de compiladores. Compilación vs. interpretación. Estructura de un compilador. Análisis léxico. Análisis sintáctico. Métodos ascendentes y descendentes. Tabla de símbolos. Árboles de parsing y árboles de sintaxis abstracta. Tratamiento de errores. Análisis semántico. Verificación de tipos. Generación de código. Optimización de código.

Arquitectura de software 2:

Escalabilidad, eficiencia y efectividad. Dimensionamiento de los requerimientos de hardware y de las necesidades de red de un sistema de software. Técnicas para escalamiento vertical y horizontal. Clustering, balanceo de carga, afinidad, sharding. Estrategias de particionamiento de bases de datos. Tolerancia a fallos. Estrategias de cache de datos. Nociones de minería de datos. Hardware específico para sistemas de gran envergadura. Virtualización. Software y hardware como servicios. Verificación del cumplimiento de los requerimientos no funcionales: performance, tolerancia a fallos, carga. Operación y monitoreo de sistemas. Estrategias de logging para sistemas de gran envergadura. Herramientas para medición de performance. Profiling. Información caliente e información de ciclo de vida largo. Análisis de servicios en red, análisis de tráfico. Herramientas de monitoreo de fallas. Metodología DevOps.

Práctica profesional supervisada:

Roles y funciones dentro de un equipo. Trabajo en equipo. Autoaprendizaje. Buenas prácticas en la gestión de recursos informáticos. Planificación de recursos. Sistemas de gestión de código integrado. Trabajo en forma remota. Relevamiento y selección de información. Conocer normas, estándares y criterios de calidad. Elaboración de Informes. Exposición de informes.

Integración en contextos reales de trabajo. Reconocimiento de roles en esos contextos. Aplicación en el marco de empresas de desarrollo de software, instituciones, organismos y sectores productivos. Vinculación con problemáticas concretas, metodologías vigentes y entornos tecnológicos reales. Procesos de acceso al mercado laboral.

Gestión de proyectos de desarrollo de software:

Planificación y estimación de proyectos de software. Definición y documentación de las actividades. Priorización y secuenciación de actividades. Dependencias, diagramas de Gantt, en iteraciones, conceptos de sprint y backlog. Asignación de recursos. Monitoreo y seguimiento de proyectos de software. Nociones de aseguramiento de calidad. Calidad de Software: del producto y del proceso. Gestión del equipo de trabajo. Herramientas de colaboración y comunicación interna. Gestión de riesgos. Seguimiento y Control de Riesgos. Gestión de la relación con el cliente. Comunicación y resolución de conflictos. Control de cambios. Mantenimiento de un producto de software.

Aprendizaje automático:

Aprendizaje, inferencia y predicción. Tipos de problemas que resuelve el aprendizaje automático (AP). Métricas de evaluación: precisión, accuracy, recall. F1-score. Matriz de confusión. Aprendizaje supervisado. Clasificadores. Árboles de decisión. Máquinas con vectores soporte. Regresión lineal, múltiple y logística.

Sistemas distribuidos y tiempo real:

Introducción a los sistemas de procesamiento distribuido y su terminología. Comunicación y sincronización en sistemas distribuidos, pasaje de mensaje y llamadas a procedimiento remoto (rpc). Tiempo, Sincronización y Coordinación Distribuida. Memoria compartida distribuida, asignación de tareas y balance de cargas (Algoritmos básicos). Manejo de Recursos y Sistemas de Archivos en Sistemas Distribuidos. Control de Concurrencia en sistemas distribuidos.

Transacciones. Seguridad en sistemas distribuidos. Computación y tecnologías verdes: eficiencia energética en sistemas.

Arquitectura de computadoras:

Arquitectura y organización de computadoras. Jerarquías de memoria: Memoria segmentada, Memoria virtual. Interrupciones: Concepto y definición. Lenguaje Ensamblador. Tipos de interrupción. Definición de entorno y contexto de un programa. Detección de interrupción: cambio de contexto. Atención de interrupciones. Subsistema de Entrada y salida. Coprocesadores (aritméticos, de video, etc). Procesadores de alta prestación. Nivel de Microarquitectura: Unidad de control, Memoria de control, Microprograma, Microinstrucciones, Cronología de microinstrucciones, Secuenciamiento de microinstrucciones. Tipos de arquitecturas: Arquitectura RISC, arquitectura en paralelo, Pentium, arquitecturas GRID, Arquitecturas multiprocesadores. Conceptos de arquitecturas basadas en servicios. Conceptos de arquitecturas reconfigurables.

Proyecto final:

Determinación de un tema de investigación y/o desarrollo: pertinencia y factibilidad. Objetivos e hipótesis. Estado del arte, revisión y búsqueda bibliográfica sistematizada. Modelización de un proyecto. Etapas de un proyecto. Estructura de escritura final y formatos de presentación.

10.1 Actividades Curriculares Acreditables (ACA)

El Instituto de Tecnología e Ingeniería definirá periódicamente el catálogo de *Actividades Curriculares Acreditables (ACA)*, que incluirá tanto unidades curriculares electivas como otro tipo de actividades académicas, investigativas, culturales, deportivas o de vinculación con la comunidad, sean estas organizadas por la UNAHUR o por otras instituciones y espacios reconocidos. También se definirán los requisitos de reconocimiento de las distintas ACA y los criterios para la ponderación y otorgamiento de créditos.

Anexo I - Contenidos mínimos de Asignaturas UNAHUR

Arte y tecnología. Escuela de espectadores

La mirada del espectador. Exploración de las múltiples conexiones que existen entre la literatura, el cine, el teatro y las artes plásticas y su relación con la tecnología. Artes plásticas. Lengua y literatura. Teatro y representación. Cine y tecnología. Fotografía.

Astro: relación de la humanidad con el cosmos

Temas y problemas de Astronomía, en una visión general, contextual e histórica. La Astronomía en la Antigüedad. La Esfera Celeste. Elementos de sistemas de coordenadas esféricas. El Tiempo Astronómico. Sistema Solar. Elementos de Astrofísica. Estrellas. Sistemas Estelares. Elementos de Cosmología. Nuevos mundos: Sistemas Extrasolares.

Ciencia, tecnología e innovación para el desarrollo

Definiciones fundamentales de ciencia, tecnología e innovación, incluyendo su importancia en el desarrollo económico y social. Los contextos y desafíos de la innovación en diversos sectores y entornos son explorados, junto con estrategias para la identificación de oportunidades y la transferencia de tecnología. Se examina el impacto ético y social de la tecnología, así como los aspectos legales y políticas públicas relacionados. Además, se fomenta el desarrollo de habilidades de trabajo en equipo y comunicación efectiva en el contexto de la innovación.

Cine documental: miradas desde el Sur

Cambios en el mundo contemporáneo y en la Argentina. El cine documental y la representación de esos acontecimientos. Las vivencias en los cambios individuales y colectivos en perspectiva de derechos humanos, de género, de nuevos hábitos y costumbres en torno al trabajo, la familia, la convivencia entre generaciones, las rupturas y los nuevos acuerdos que se producen entre jóvenes y adultos en relación con la forma de entender el mundo contemporáneo. El documental y la representación de "la justicia" y sus instituciones en el cine nacional. El documental y la representación de "la justicia" y sus instituciones en el cine internacional.

Ciudadanía activa y compromiso social

Las políticas de infancias, el rol del Estado y las nuevas prioridades de agenda en derechos de la niñez y en la reducción de las desigualdades en la Argentina y en el contexto latinoamericano. Las políticas sociales de infancias, la igualdad de oportunidades y de resultados; los paradigmas de políticas de infancia y adolescencia en Argentina y América Latina; la desigualdad y la pobreza en la infancia y adolescencia; la inversión social.

Cuando los pasados no pasan: lugares de memoria

La memoria. La noción de “lugares de memoria”. Genocidios del siglo XX: un acercamiento histórico y conceptual. El terrorismo de Estado en Argentina. Políticas de memoria: derechos humanos ayer y hoy. El memorial de Berlín; la historia de vida de Soghomon Tehlirian; la fecha del 24 de marzo; el pañuelo de las Madres; el Himno Nacional Argentino o el Museo/sitio de memoria ESMA pensados críticamente para conocer el pasado y construir una economía general del pasado en el presente.

Debates políticos actuales. Ideas para pensar el mundo de hoy

Introducción al debate político. El debate político contemporáneo y las singularidades del momento histórico-ideológico actual. Algunos debates políticos actuales, tales como la justicia social, la igualdad de género, la ecología, el avance tecnológico, el populismo y la antipolítica.

Educación sexual integral. Cuando lo esencial es visible a los ojos

Introducción a la Educación sexual integral: enfoques y tradiciones de la educación sexual. El paradigma de derechos como marco para las prácticas pedagógicas de ESI: Declaración de los Derechos Humanos y otras leyes que cambiaron paradigmas. La Ley Nacional N° 26.150/06. Nuevas/os sujetos/as: niñez y adolescencia; autonomía progresiva; superación del paradigma tutelar. Educación Sexual Integral con perspectiva de género. Géneros y diversidades. El cuerpo como construcción política.

Filosofía. Problemas filosóficos

Orígenes de la Filosofía: Grecia. La filosofía entre el arte y la ciencia. La pregunta por el todo. La duda radical. Definiciones críticas de la filosofía. El poder. La multiplicidad de relaciones de poder. El poder y el discurso. La voluntad de poder. Posmodernidad y la sociedad del espectáculo. El fin de los grandes relatos. El cuestionamiento de la idea de progreso y de la teleología de la historia.

Posmodernidad y posverdad, sociedad de la comunicación, sociedad de consumo, sociedad del espectáculo. El otro. Existencia precaria y política. La idea de libertad y la ética de la responsabilidad. El debate en torno a los conceptos de tolerancia y hospitalidad. El extranjero.

Innovación y creatividad

Creatividad, e innovación. La innovación y el desarrollo en los campos del conocimiento asociados a las especialidades o de las carreras de la Unahur. El contexto sociocultural de la innovación. ¿Para quiénes innovamos desde la Universidad? Proceso creativo. Diagnóstico de la problemática. Técnicas de generación de ideas. Nociones básicas de neuroeducación para aplicarlas a la generación de ideas-proyecto. Innovación Social Sustentable. Nuevos modelos de liderazgo. Conceptos y desarrollo. Difusión. Formas de organización. Apoyo y financiamiento. Modelos de inversión actuales. Modelos de presupuesto. Financiamiento. Innovación Colaborativa. Organización. Modelo Canvas. Cómo cuento mi proyecto. Cómo muestro mi proyecto.

Introducción a la imagen. De la imagen fija a la imagen en movimiento

Enfoque semiótico y giro pictórico. El problema de la representación. La imagen como signo. La relación entre el significado y el referente. El lenguaje de los nuevos medios. La cultura visual y el estudio de la visualidad. La imagen mediática. La retórica de la imagen. El acto fotográfico. La potencia política de las imágenes. Collage y montaje. El lugar del espectador emancipado. Herramientas del lenguaje visual. Artes y medios visuales y audiovisuales. La estética de lo performativo y la teatralidad.

Introducción al Latín

Nociones básicas sobre los orígenes de la lengua latina. El latín y las lenguas romances. la vida cotidiana en Roma. Epitafios y graffitis. La construcción de la identidad romana. La condición de la mujer en la antigüedad latina. Palabras flexivas. Morfología nominal. Hechiceras, magas y diosas en la cultura latina. Representaciones para la mujer en la tragedia latina. La puella culta elegíaca. Su contexto de aparición: una nueva manera de ser mujer en Roma.

La vida secreta de las rocas

Introducción a la geología: origen y evolución del universo, el Sistema Solar y la Tierra. El tiempo geológico. Introducción a la paleontología: evolución e historia de

la vida en la Tierra. Registro geológico. Cambio climático. Mineralogía: propiedades de los minerales. Métodos de identificación de minerales. Introducción a la sistemática mineral. El ciclo de las rocas: Procesos endógenos y exógenos. Geología e hidrocarburos: Sistema petrolero convencional y no convencional. Importancia estratégica e implicancias ambientales de las actividades.

Malvinas: una causa de nuestra América Latina

Los principales argumentos históricos. Descubrimiento, colonización y usurpación. Los argumentos jurídicos: de la usurpación a las Naciones Unidas. Malvinas como causa política de Estado. Integridad territorial y Libre determinación de los pueblos. Otros casos de colonialismo bajo la bandera de la libre determinación.

El Atlántico Sur en la geopolítica de América Latina: recursos naturales, depredación y militarización. Soberanía sobre el Atlántico Sur. La Antártida como espacio de disputa.

Historia contemporánea de la causa Malvinas: guerra y posguerra. Inglaterra y los problemas de financiamiento de las islas. Intercambios en materia de comunicación, recursos energéticos y educación. El golpe cívico militar de 1976 y el cambio de perspectiva. La decisión de tomar Malvinas y la derrota. Los ochenta y los noventa: la “desmalvinización”. Posneoliberalismo y remalvinización. Malvinas como causa regional. Un nuevo período de desmalvinización.

Métodos participativos de transformación de conflictos

El diálogo colaborativo y la construcción de consensos. Convivencia ambiental. Teoría del Conflicto. Su apreciación y tratamiento como oportunidad de cambio. Comunicación. Conocimientos básicos y aplicación a la vida comunitaria y profesional. Negociación. Técnicas y herramientas. Mediación. Procesos de mediación y su incidencia en la cultura. Facilitación en procesos de abordaje de conflictos intra e inter institucionales. Procesos participativos de prevención temprana y adecuado abordaje de conflictos comunitarios.

Pensamiento ambiental latinoamericano

Introducción al pensamiento ambiental latinoamericano (PAL). La educación y el desarrollo como dos ejes y preocupaciones centrales del PAL. El rol de la educación superior: avances y desafíos. Las concepciones del desarrollo que se disputan al Norte global. La incorporación de la dimensión ambiental en la educación superior. De la EA a la Educación para el Desarrollo Sustentable: un desplazamiento que no

sólo es conceptual sino político. El postdesarrollo como alternativa al desarrollo. La ecología política y la propuesta de descolonizar la naturaleza.

Robótica

Tipos de robots y campos de uso. Partes que componen un dispositivo robótico. Conceptos de tinkercad y su uso. Conceptos básicos de arduino. Algebra de Boole y lógica digital. Introducción a la programación en bloques y C++. Robótica y automatización de objetos.

Una historia del rock nacional

Los orígenes del Rock Nacional. Las derivas urbanas como método compositivo. El núcleo fundador. Espacios de sociabilidad. La jerga del rock. Rock y marginalidad. El Cordobazo. La década del 70. Inspiraciones bajo el látigo de la violencia. El apogeo del Rock Nacional. Concepto de "música progresiva". Folklore y rock. El rock sinfónico. La década del 80. Modernidad o muerte. La guerra de Malvinas como separatoria de aguas. La recuperación democrática. La rebelión punk. De los teatros y estadios al pub y los lugares emblemáticos. El canto popular urbano. La década del 90. La balsa a la deriva. La canción neoliberal. Año 2000 y después. La vuelta de Boedo y Florida: la movida sónica y el rock chabón. Las tribus urbanas. Experimentación y poesía social. Cumbia y protesta social. Últimos años: La producción independiente y las nuevas tecnologías. La muerte del disco.