

Programa de Cátedra

	Asignatura:	Departamento:
	Bloque: Disciplinas Tecnológicas	Área: Laboratorio de Computación I
	Régimen: Primer Semestre	Horas semanales: 6
	Tipo: Troncal	Horas semestrales/anuales: 90
	Carrera: Tecnicatura Universitaria en Programación	Nivel (Año):
	Ciclo lectivo: 2021	<input checked="" type="checkbox"/> 1° <input type="checkbox"/> 2° <input type="checkbox"/> 3° <input type="checkbox"/> 4° <input type="checkbox"/> 5° <input type="checkbox"/> 6°

Integrantes de la Cátedra:

- **Profesor Titular:**

Cristina Domizio

- **Profesor Asociado:**

Gerardo Magni

- **Profesor/es Adjunto/s:**

Nombre del Profesor	Tipo de Dedicación	Cantidad de Dedicaciones
Ninguno		

- **Auxiliares de Docencia:**

Nombre del Profesor	Categoría	Tipo de Dedicación	Cantidad de Dedicaciones
Ninguno			

a) Fundamentación de la materia dentro del plan de estudios

La estructura curricular del espacio bajo análisis dispone de 6 horas-cátedra semanales. Se propone un aula-taller que básicamente persigue el objetivo de incrementar en el alumno su natural habilidad para utilizar el pensamiento elaborado lógicamente a fin de resolver problemas a través de la descomposición de los mismos, en sub-problemas más simples (diseño descendente), estableciendo la relación entre el todo y las partes (enfoque sistémico), considerando sólo los

aspectos relevantes (abstracción), mediante el uso de herramientas fundamentales representativas de los procesos deductivos asociados a la programación, integrando la sintaxis elemental de un lenguaje de programación en Laboratorio. A su vez se intenta modelar un ambiente creativo de niveles crecientes de realismo respecto del mundo laboral pertinente. Esto nos lleva a establecer la metodología de carpeta abierta y uso de documentación en soportes y en línea.

En definitiva el espacio curricular es uno de los más propicios para lograr “El desarrollo del pensamiento reflexivo, del juicio crítico, de la creatividad y de la capacitación para el auto-aprendizaje”, y a la vez, propiciar el derecho del alumno a recibir “formación científica, profesional, humanística y técnica en el más alto nivel, contribuir a la preservación de la cultura nacional, promover la generación y desarrollo del conocimiento en todas sus formas, y desarrollar las actitudes y valores que requiere la formación de personas responsables, con conciencia ética y solidaria, reflexivas, críticas, capaces de mejorar la calidad de vida, consolidar el respeto al medio ambiente, a las instituciones de la República y a la vigencia del orden democrático”.

Es obvio, luego, el rol de auxiliar y complementaria de esta asignatura, posibilitando un enfoque más práctico y orientado al uso de herramientas presentes en un entorno integrado de desarrollo (IDE por sus siglas en inglés).

Laboratorio I permite desarrollar estrategias y codificación de problemas computacionales utilizando lenguajes de programación.

El desarrollo e implementación de distintos algoritmos permite lograr la primera aproximación del programador a situaciones reales de experiencia que, en un futuro, le permitirán desarrollar algoritmos más complejos y orientados a la lógica de negocios que necesite resolver.

b) Objetivos de la materia

• **Objetivos Generales:**

La cátedra se enfoca en los siguientes objetivos generales:

- Resolver problemas complejos de programación y la arquitectura aplicada para la mejor solución.
- Conocer tecnologías, paradigmas, metodologías y herramientas de última generación que se aplican en el desarrollo de software.
- Motivar las buenas prácticas de desarrollo en función de los estándares de calidad.
- Relacionar los saberes adquiridos en análisis y diseño de sistemas, como así en programación funcional y de objetos, adaptándolos a las necesidades solicitadas para un perfil de desarrollador Full Stack.
- Conocer herramientas de productividad que acompañan el proceso de desarrollo de software.

• **Objetivos Específicos:**

La cátedra se enfoca en varios objetivos a conseguir por parte del alumno:

- Buena interpretación de las consignas propuestas para resolver mediante la programación.
- Elaborar soluciones creativas a las consignas propuestas.
- Adquirir principios de buen diseño de software.
- Organización y planificación del trabajo, para la resolución de problemas complejos.
- Desarrollo de la comunicación, de la curiosidad investigativa y del trabajo en grupo.

- Aplicar la mejor arquitectura de sistemas que se adecue a la solución solicitada.
- Determinar el correcto uso de patrones para el desarrollo de software en el Front End y Backend.
- Seleccionar el mejor esquema de persistencia para brindar la solución más efectiva.

c) Contenidos Mínimos ()

a. Algoritmos: Programa

- Estilos de programación (Estructurada – Orientada a objetos).
- Estructura de un programa Java.
- Tipos de datos en Java.
- Secuencia: Declaración de variables.
- Métodos. Instrucciones en Java.
- Decisiones: Simples. Alternativas. Múltiples. Instrucciones en Java.
- Iteraciones: Bucles. Anidamientos.
- Array: Asignación. Almacenamiento . Recorrido.
- Ordenamiento. Búsquedas.
- Conceptos Avanzados: Recursividad.
- Introducción a la programación orientada a objetos. Noción de clases.
- Introducción a listas, pilas, colas.
- Conceptos generales: Entorno de desarrollo IDE. Depuración. Stepping. Break-point. Librerías. Packages.
- Apoyo práctico a la cátedra Programación I.

d) Programa Analítico

UNIDAD TEMÁTICA	CONTENIDOS	Referencia bibliografía (Nro.)
1 Algoritmos	<p>Objetivo : Conocer las herramientas utilitarias que se usan en un Software Factory y las distintas plataformas que se usan en el desarrollo de software.</p> <p>Contenido :</p> <ol style="list-style-type: none"> 1. Problema. Contexto. Datos asociados. 2. Comprensión de problemas y metodología general de resolución. 3. Nociones de acción, estado y secuencia. 4. Algoritmo. Concepto. Elementos. Características. Tipos de algoritmos. 5. Programa. Concepto y características. Instrucción. Elementos. Constantes y variables simples. Tipos de datos. 6. Conocimiento del entorno de desarrollo. 	Apuntes de cátedra Y 3

UNIDAD TEMÁTICA	CONTENIDOS	Referencia bibliografía (Nro.)
	7. Expresiones numéricas y alfanuméricas. Expresiones condicionales: relacionales y lógicas. Tablas de verdad.	
2 Estructuras secuenciales	<p>Objetivo : Establecer los conocimientos necesarios de las distintas estructuras de programación secuenciales y de operaciones, requeridas en un programa.</p> <p>Contenido :</p> <ol style="list-style-type: none"> 1. Conceptos. 2. Representación gráfica. 3. Declaración de variables. Asignación, lectura y escritura. 4. Pre-condiciones (estado inicial requerido) y pos-condiciones (estado final provocado). 	2 - 3
3 Estructuras de decisión.	<p>Objetivo : Establecer y asegurar la estructura de la toma de decisiones dentro de un programa, y asegurar el funcionamiento correcto para cada caso particular, analizando el dato conocido y el desconocido. Armado correcto de afirmaciones lógicas.</p> <p>Contenido :</p> <ol style="list-style-type: none"> 1. Concepto. 2. Representación gráfica con diagramación de flujo y estructurada (Chapín). 3. Composición condicional (decisión simple) y alternativa (decisión doble). Decisiones múltiples (alternativas anidadas y composición selectiva). 4. Pre y pos-condiciones de cada una. 	2 – 3 - 4
	<p>TECNOLOGIAS DE PROGRAMACION DEL LADO DEL SERVIDOR</p> <p>Objetivo : Establecer el conocimiento del funcionamiento de la estructura repetitiva, estableciendo la condición adecuada según la necesidad para el cumplimiento del objetivo final.</p>	6,7

UNIDAD TEMÁTICA	CONTENIDOS	Referencia bibliografía (Nro.)
4 Estructuras cíclicas	<p>Contenido :</p> <ol style="list-style-type: none"> 1. Conceptos. Representación gráfica. 2. Bucles lógicos: Control de inicio de Bucle (Mientras-Fin Mientras, Para- Fin Para) y Control de fin de Bucle (Repetir – Hasta que). 3. Pre y pos-condiciones de cada una de ellas. Definición de invariante de un bucle. 	2 -3 -4 -5
5 Recursión	<p>Objetivo : Conocer y construir la estructura general de un programa, y lograr agrupar acciones repetitivas en bloques, para que sean llamados en forma recursiva, evitando el código redundante y logrando la robustez y la reducción de código innecesario.</p> <p>Contenido :</p> <ol style="list-style-type: none"> 1. Subprogramas: concepto, argumentos, parámetros. 2. Pasaje de argumentos por valor y por referencia. 3. Noción de llamada a procedimientos y funciones. 	1- 2- 3- 4- 5
6 Variables dimensionadas	<p>Objetivo : Conocer y construir la variable con una o más dimensiones. Utilizar la dimensión para mejorar la performance del programa, Determinar la necesidad de la dimensión para agilizar el funcionamiento y la efectividad de los resultados, basándose en los índices como iniciación de su utilización.</p> <p>Contenido :</p> <ol style="list-style-type: none"> 1. Concepto. Dimensión e índices. 2. Declaración. Inicialización. Utilización. 3. Operaciones asociadas. 4. Ordenamiento y Búsqueda. 5. Variables multi-dimensionadas. 	1 – 2 – 3 – 4 – 5 – 6 – 7 - 8
7 Colecciones	<p>Objetivo : Las colecciones son una especie de arrays de tamaño dinámico. Cuando queremos trabajar con un conjunto de elementos, necesitamos un almacén donde poder guardarlos. En Java, se</p>	

UNIDAD TEMÁTICA	CONTENIDOS	Referencia bibliografía (Nro.)
	<p>emplea la interfaz genérica Collection para este propósito. Gracias a esta interfaz, podemos almacenar cualquier tipo de objeto y podemos usar una serie de métodos comunes, como pueden ser: añadir, eliminar, obtener el tamaño de la colección, etc.</p> <p>Contenido :</p> <ol style="list-style-type: none"> 1. Concepto. 2. Declaración. Inicialización. Utilización. 3. Operaciones asociadas. 4. Collection<E> 5. List<E> (ArrayList<E>, Vector<E>, LinkedList<E>) 6. Set<E> 7. Map<K,V> 	

e) Programa de Examen

Idem programa Analítico.

f) Trabajos Prácticos

Unidad a la que corresponde	Título del trabajo práctico/actividad de laboratorio/taller/etc.	Objetivo	Temas a aplicar/cubrir según programa (Nro.)
1	Práctico 1: Toma de decisiones	Construir afirmaciones lógicas con toda clase de operadores	1.1 al 1.7 1.2
2 - 3	Práctico 2. : Decisiones y Menú	Atribuir una actividad diferenciada según la toma de decisiones	2.1 al 2.4 3.1 al 3.4
4	Práctico 3: Bucles	Identificar acciones repetitivas dentro del código.	4.1 al 4.3
3 - 4	Práctico 4: Diagramación Lógica y	Realizar instalación y configuración	3.2, 4.2 y 4.3

	estructurada	avanzada del archivo POM	
5	Práctico 5: Procedimientos y funciones	Identificar acciones repetitivas dentro del código, y construir código independiente, determinando datos conocidos y desconocidos.	5.1 al 5.3
6	Práctico 6: Variables dimensionadas	Determinar la dimensión necesaria para variables, para mejorar el rendimiento del código	6.1 al 6.5
6	Práctico 7: Ordenamiento	Conocer los métodos existentes necesarios para el ordenamiento de variables dimensionadas	6.3 y 6.4
7	Práctico 8: Colecciones	Aplicar colecciones para la manipulación dinámica de datos.	7

g) Distribución de horas

Formación teórica	30
Formación experimental	30
Resolución de problemas de ingeniería	15
Proyecto y diseño	15

h) Correlativas

Para cursar		Para rendir
Tener regulares	Tener Aprobadas	Tener aprobadas
<ul style="list-style-type: none"> Ninguna 	<ul style="list-style-type: none"> Ninguna 	<ul style="list-style-type: none"> Ninguna

i) Bibliografía

Nro.	Autor/es	Título	Editorial	Año de edición	Principal	Complementaria
1	Deitel & Deitel	Java How to Program 7/e	Prentice Hall.	2008		x
2	Fco. Javier Ceballos	JAVA 2: Curso de Programación	Alfaomega-Rama		x	
3	Docente	Apuntes entregados			x	
4	Patrick Naughtn	Manual de JAVA	MC Graw Hill		x	
5	F Lima Diaz	Manual Avanzado de Java	Anaya			x
6	Thomas Wu	Introducción a la Programación Orientada a Objetos con Java	Mc Graw Hill			x
7	Bruce Eckell	Pensando en Java	Prentice Hall.			x
8	Patrick Naughtn	Manual de Java	Mc Graw Hill			x

10- SITIO WEB OFICIAL

- Java - Oracle: <https://www.java.com/es/>
- NetBeans: <https://netbeans.org/>

Nombre del director	Nombre del encargado de la Cátedra
Ing. Osvaldo Giordaninni	Cristina Domizio, Gerardo Magni
Firma del Director	Firma del encargado de la Cátedra
Fecha de entrega del programa	Febrero de 2020

Planificación de Cátedra

a) Metodología de Enseñanza

La estructura curricular del espacio bajo análisis dispone de 6 horas-cátedra semanales. Se propone un aula-taller que básicamente persigue el objetivo de incrementar en el alumno su natural habilidad para utilizar el pensamiento elaborado lógicamente a fin de resolver problemas a través de la descomposición de los mismos, en subproblemas más simples (diseño descendente), estableciendo la relación entre el todo y las partes (enfoque sistémico), considerando sólo los aspectos relevantes (abstracción), mediante el uso de herramientas fundamentales representativas de los procesos deductivos asociados a la programación, integrando la sintaxis elemental de un lenguaje de programación en Laboratorio. A su vez se intenta modelar un ambiente creativo de niveles crecientes de realismo respecto del mundo laboral pertinente. Esto nos lleva a establecer la metodología de carpeta abierta y uso de documentación en soportes y en línea.

En definitiva el espacio curricular es uno de los más propicios para lograr “El desarrollo del pensamiento reflexivo, del juicio crítico, de la creatividad y de la capacitación para el auto-aprendizaje”¹, y a la vez, propiciar el derecho del alumno a recibir “formación científica, profesional, humanística y técnica en el más alto nivel, contribuir a la preservación de la cultura nacional, promover la generación y desarrollo del conocimiento en todas sus formas, y desarrollar las actitudes y valores que requiere la formación de personas responsables, con conciencia ética y solidaria, reflexivas, críticas, capaces de mejorar la calidad de vida, consolidar el respeto al medio ambiente, a las instituciones de la República y a la vigencia del orden democrático”².

Es obvio, luego, el rol de auxiliar y complementaria de esta asignatura, posibilitando un enfoque más práctico y orientado al uso de herramientas presentes en un entorno integrado de desarrollo (IDE por sus siglas en inglés).

Finalmente, debemos destacar que, además de ser la programación uno de los campos más específicos, aplicables y solicitados dentro del mundo laboral, es a la vez, claramente influyente en la corroboración vocacional del educando. Así el mismo disfrutará (o padecerá) intensamente el exigente cursado de esta asignatura.

¹Ley 6970, Título II: Principios y Fines de la Educación, Cap II: Fines de la Educación, Art. 6°, inc. C.

²Ley 24521, Título II: Capítulo 1: De los Fines y Objetivos, Art. 3°.

La metodología de enseñanza de esta materia es teórico-práctica, el abordaje de los temas por parte de los profesores invita a los alumnos a la realización continua de ejercicios prácticos

Los docentes guían y motivan a los alumnos en la formación de equipos de trabajo, tratando de lograr la homogeneidad de los integrantes respecto de carga de materias, disponibilidad de tiempo, conocimientos y experiencia laboral, a fin de disponer en el curso de equipos con similares condiciones.

Cada alumno debe tener la carpeta de trabajos prácticos completa, para su trabajo y a disposición de los docentes, en todas las clases. Para tal fin se utilizará el aula virtual.

Metodología	Explicación de la modalidad
Clases teóricas	<ul style="list-style-type: none">. Explicación interactiva del docente.. Discusiones entre equipos de trabajo.. Autoaprendizaje e interpretación de textos.
Trabajos Prácticos	<ul style="list-style-type: none">. Discusiones entre equipos de trabajo.. Investigación y presentación de resultados.
Prácticas en laboratorio	<ul style="list-style-type: none">. Análisis y resolución de casos en clase.. Presentación de experiencias y conocimientos.
Otras	

b) Cronograma de actividades

Semana N°	Unidad	Contenidos	Objetivos	Actividades	Evaluaciones	Recursos	
						Bibliográficos	Didácticos
1	1	Introducción a la programación – Introducción a Java- Máquina virtual – Editores – Intérpretes– Compiladores. Programa fuente – Bytecode - Package Tipos de datos – Definición de variables	Conocer los elementos esenciales para el funcionamiento y el aprovechamiento de la programación	Ejercicios en clase. TP 1		3 Apuntes de cátedra	PowerPoint Laboratorio Aula Virtual Pizarra
2	1 2	Introducción a la Ide de programación Netbeans – Conceptos de proyecto, paquetes Clases principal, método main - Estructura de un programa en java – Estructuras secuenciales - Introducción a métodos estáticos – Ejercitación.	Probar todo lo teórico en ejercicios utilizando el entorno en pantalla. Confirmación de resultados y búsqueda de mejores caminos para la solución de problemas	Desarrollo teórico y práctico TP:2		Apuntes de cátedra	PowerPoint Laboratorio Aula Virtual Apuntes de cátedra Internet
3	3	Sentencias condicionales –	Construcción de expresiones lógicas	Desarrollo teórico y práctico.		Apuntes de cátedra	PowerPoint

		Condición simple – Condición compuesta - Anidamiento	simples y compuestas. Búsqueda de soluciones según el resultado lógico. Conocimiento del menú	Ejercicios en clase. TP: 3			Laboratorio Aula Virtual Apuntes de cátedra Internet
4	4	Iteraciones – Ciclo for – While –	Valorar la importancia de las condiciones en los bucles.	Desarrollo teórico y práctico Ejercicios en clase TP:3		Apuntes de cátedra 2 – 3 - 4	PowerPoint Laboratorio Aula Virtual Apuntes de cátedra Internet
5	4	Iteraciones - Do while	Desarrollo de código acompañado con diagramación lógica y estructurada. Conocimiento de la simbología. Anidamientos	Desarrollo teórico y práctico TP:4		Apuntes de cátedra 2 – 3 - 4	PowerPoint Laboratorio Aula Virtual Apuntes de cátedra Internet
6	4	Corte de control simple – Corte de control compuesto	Conocer el mecanismo para el control de los bucles según distintas	Desarrollo teórico y práctico. Ejercicios en clase TP:4		Apuntes de cátedra	PowerPoint Laboratorio Aula Virtual

			herramientas de programación			2 – 3 - 4	Apuntes de cátedra Internet
7	1-2-3-4	Evaluación parcial y entrega de la carpeta de trabajos prácticos. Elaboración de ejercicios	Control de conocimientos y destrezas adquiridos por el alumno en esta primera etapa.	La evaluación puede ser por plataforma, en forma escrita en papel, o en procesador de texto, o bien un trabajo entregado. Es conveniente que sea individual. No grupal.	Primera evaluación	Apuntes de cátedra 2 – 3 - 4	PowerPoint
8	5	Uso de subprogramas. Comparación con funciones existentes en el lenguaje.	Creación y uso de funciones nuevas.	Desarrollo teórico y práctico. Ejercicios en clase TP:5		Apuntes de cátedra 2 – 3 - 4	PowerPoint Laboratorio Aula Virtual Apuntes de cátedra Internet
9	6	Ejercitación de arreglos unidimensionales	Comprender y reconocer el papel importante del índice.	Desarrollo teórico y práctico. Ejercicios en clase TP:6		Apuntes de cátedra 2 – 3 – 4	PowerPoint Laboratorio Aula Virtual Apuntes de cátedra Internet
10	6	Introducción a matrices. Declaración –	Comprender y reconocer el papel importante del	Desarrollo teórico y práctico. Ejercicios en clase		Apuntes de cátedra	PowerPoint Laboratorio

		inicialización – Carga.	índice, con más de una dimensión.	TP:6		2 – 3 – 4	Aula Virtual Apuntes de cátedra Internet
11	6	Métodos de ordenamiento – Buble sort – Selection sort – Insertion sort – Shell sort – Heap sort – merge sort – Quick sort .	Comprender los mecanismos de ordenamiento de colecciones de datos. Uso de intercambio de valores.	Desarrollo teórico y práctico. Ejercicios en clase TP: 7		Apuntes de cátedra 2 – 3 – 4	PowerPoint Laboratorio Aula Virtual Apuntes de cátedra Internet
12	6	Ejercitación con métodos de ordenamiento y algoritmos de búsquedas..	Repaso general de todos los temas aplicados a ordenamiento y/o búsqueda en arreglos de una y más dimensiones.	Desarrollo teórico y práctico. Ejercicios en clase TP:7		Apuntes de cátedra 2 – 3 – 4	PowerPoint Laboratorio Aula Virtual Apuntes de cátedra Internet
13	7	Ejercitación haciendo uso de Colecciones	Comprender las ventajas de las estructuras de datos dinamicas	Desarrollo teórico y práctico. Ejercicios en clase TP:8		Apuntes de cátedra	PowerPoint Laboratorio

							Aula Virtual Apuntes de cátedra Internet
14	1 – 2 – 3 – 4 – 5 – 6 – 7	Examen global	Control de la totalidad de conocimientos y destrezas adquiridos por el alumno.	La evaluación puede ser por plataforma, en forma escrita en papel, o en procesador de texto, o bien un trabajo entregado. Es conveniente que sea individual. No grupal.		1 – 2 – 3 – 4 – 5 – 6 – 7	
15	1 – 2 – 3 – 4 – 5 – 6 – 7	Recuperatorio del examen global	Control de la totalidad de conocimientos y destrezas adquiridos por el alumno.	La evaluación puede ser por plataforma, en forma escrita en papel, o en procesador de texto, o bien un trabajo entregado. Es conveniente que sea individual. No grupal.		1 – 2 – 3 – 4 – 5 – 6 – 7	

c) Trabajos de campo, visitas a empresas

Actividad	Objetivo	Lugar	Responsable	Evaluación
Trabajo realizado en grupo, investigando las estructuras de programas existentes en oficinas públicas y privadas.	Motivar al alumno en las nuevas tecnologías de desarrollo.	Empresas de Mendoza elegidas por el alumno.	Docente - alumnos	No

d) Articulación horizontal y vertical con otras materias

Esta materia está relacionada con todas las áreas, ya que al momento de desarrollar una aplicación, se debe considerar cualquier aspecto de la realidad que lo requiera.

La articulación se obtiene a través de:

- Resolución de ejercicios citando ejemplos de toda clase de ámbitos.
- Utilizar la investigación para mostrar claramente la necesidad del dominio del idioma inglés, facilitando un mejor desempeño y resultados.
- Utilización de las matemáticas para facilitar los razonamientos.
- Los diversos desarrollos conllevan la necesaria investigación y acatamiento de normas y leyes, en el trato de la información.

Desarrollos propuestos para lograr la articulación:

- Presentación de experiencias reales que el Docente expone, explica y somete a consideración de los alumnos.
- Trabajos prácticos que permiten al alumno realizar una práctica integrada, recuperando los conocimientos recibidos en las Asignaturas de la carrera hasta el momento.
- Casos con situaciones, condiciones y escenarios simulados por el Docente, que los alumnos deben resolver individualmente o en equipo.

Articulación con la Materia:	Nivel (Año de la Carrera)
Programación 1	Año 1 – Primer semestre
Matemática	Año 1 – Primer semestre
<i>Inglés 1</i>	Año 1 – Primer semestre
Sistemas de Procesamiento de datos	Año 1 – Primer semestre
Legislación	Año 2 – Cuarto semestre

		Materia relacionada								
Temas de la Cátedra	Temas	Programación 1	Laboratorio computación 1	Matemática	Inglés 1	Sistema de Procesamiento de Datos	Legislación			
Laboratorio de Entornos de Desarrollo y Herramientas		x	x			x				
Armado de afirmaciones lógicas		x	x	x		x				
Investigación y resolución de problemas		x	x	x	x	x				
Interoperabilidad entre sistemas		x	x	x	x	x	x			
Desarrollo de sistemas		x	x	x	x	x	x			

e) Régimen de cursado y aprobación

1) Aspectos considerados en la evaluación.

Instancias evaluativas parciales:

- Primer examen parcial.
- Examen global (que comprende también los temas del primer parcial).
- Examen recuperatorio del Examen Global. Rinden alumnos que no hayan obtenido la aprobación no directa (regular) o aquellos que estando en condiciones de aprobar en forma no directa aspiran a la aprobación directa (promoción).

2) Forma de evaluación y controles.

- Ambos exámenes son teórico - prácticos. La forma de evaluación es escrita e individual pudiendo usar también para ambas instancias el aula virtual. Esta evaluación formal consiste en una prueba de conocimientos y aplicación de estos.
- Examen recuperatorio es teórico práctico para los alumnos que aspiran a la aprobación directa (promoción) y práctico para los alumnos que aspiran a la aprobación no directa (regular).

3) Instancias de aprobación

Condiciones para obtener la APROBACIÓN DIRECTA:

Los alumnos obtendrán la Aprobación Directa con 6 o más (60% o más), Aprobación No Directa con 4 ó 5 (40% a 59%) y No Aprobación con 1, 2 y 3 (1 a 39).

También deben considerar que aprobado es con 60% ó más para los exámenes finales.

Condiciones para obtener la APROBACIÓN NO DIRECTA:

Todos los que no alcancen del 60% en ambos parciales, pueden recuperar, el examen global, para alcanzar la Aprobación Directa o la No Directa.

4) Instancias de recuperación

- El examen global puede ser recuperado por aquéllos alumnos que obtuvieron una calificación menor al 60% tanto en el primer parcial como en el global. En cada instancia de recuperación se mantiene la condición del alumno lograda a ese momento.

Si el alumno pasa todas las instancias de recuperación requeridas para llegar a la regularidad de la materia y en algunas de éstas tiene aplazo, queda Libre.

Aprobación Directa: SI

Notas, valores o % mínimos esperados	Para aprobar en forma directa		Para aprobar en forma no directa	
	Individual	Grupal	Individual	Grupal
Parciales	60%		40% o 50%	
Examen global Integrador	60%		40% o 50%	
Recuperatorio global Integrador	60%		40% o 50%	
Otros:				
Asistencia (mínimo 75%).	75%		75%	

5) Modalidad de examen final

- Evaluación teórica práctica de:
 - . Trabajos prácticos Individuales Nro. 1 al 6
 - . Temas teóricos tratados durante el cursado

f) Actividades del equipo docente

Docente	Categoría	Dedicación	Actividades			
			Docencia	Gestión	Investigación	Extensión
Domizio, Cristina	Titular	1	Responsable de la cátedra			
Magni, Gerardo	Adjunto	1	Responsable de la cátedra			

g) Observaciones

Se evalúa con un examen parcial, y luego con un examen global debido a que no es posible separar los temas impartidos antes de la primera instancia de evaluación. En la segunda instancia, se agregan los temas impartidos, posteriores a la primera evaluación, sin poder separarlos. De los antes mencionados.

h) Horario de Consulta de Profesores y Auxiliares

- Primer Semestre

Nombre del docente: Domizio, Cristina

Martes: 19 a 20hs.

Nombre del docente: Magni Gerardo

Viernes: 19 a 20hs.

- Segundo Semestre

Nombre del docente: Domizio, Cristina

Martes: 19 a 20hs.

Nombre del docente: Magni Gerardo

Viernes: 19 a 20hs.

Nombre del director	Nombre del encargado de la Cátedra
Ing. Osvaldo Giodaninni	Domizio, Cristina Graciela Magni Gerardo
Firma del Director	Firma del encargado de la Cátedra
Fecha de entrega de la planificación	15 de febrero de 2020