

MODALIDAD ACADÉMICA

Asignatura	Tecnología de Software de Base – Electiva Tercer Nivel	
Carrera	INGENIERÍA EN SISTEMAS DE INFORMACIÓN	
Ciclo Lectivo	2017	
Vigencia del programa	Desde el ciclo lectivo 2017	
Plan	2008	
Nivel	<input type="checkbox"/> 1er. Nivel <input type="checkbox"/> 2do. Nivel <input checked="" type="checkbox"/> 3er. Nivel <input type="checkbox"/> 4to. Nivel <input type="checkbox"/> 5to. Nivel	
Coordinador de la Cátedra	Ing. Valerio Fritjelli	
Área de Conocimiento	<input checked="" type="checkbox"/> Programación <input type="checkbox"/> Computación <input type="checkbox"/> Sistemas de Información <input type="checkbox"/> Gestión Ingenieril <input type="checkbox"/> Modelos <input type="checkbox"/> Complementaria	
Carga horaria semanal	8 horas cátedra.	
Anual/ cuatrimestral	Cuatrimestral	
Contenidos Mínimos, según Diseño Curricular-Ordenanza 1150 (sólo para asignaturas curriculares)	No aplica.	
Correlativas para cursar (según Diseño Curricular-Ordenanza 1150)	Regulares	Aprobadas
	<ul style="list-style-type: none"> Paradigmas de Programación 	<ul style="list-style-type: none"> Algoritmos y Estructuras de Datos
Correlativas para rendir (según Diseño Curricular-Ordenanza 1150)	Regulares	Aprobadas
	<ul style="list-style-type: none"> Paradigmas de Programación 	<ul style="list-style-type: none"> Paradigmas de Programación
Objetivos de la Asignatura	<p><i>Que el estudiante:</i></p> <ul style="list-style-type: none"> Comprenda, defina y aplique estructuras de datos avanzadas, junto con los algoritmos asociados a ellas, para la resolución de problemas y situaciones complejas. Incorpore elementos de análisis y diseño de algoritmos, para medir la eficiencia relativa de distintos algoritmos que pudieran servir para el mismo problema. Implemente eficientemente las soluciones planteadas, aplicando buenas prácticas y patrones de diseño, mediante una plataforma de programación de uso profesional actual, para facilitar la futura inserción del estudiante en el mundo laboral. 	
<u>Programa Analítico</u>		
Unidad Nro. 1: Fundamentos de Programación Orientada a Objetos en Java.		
Objetivos específicos:		

Que el estudiante repase, adquiera y/o profundice conceptos y técnicas de programación orientada a objetos en Java, así como los fundamentos del propio lenguaje Java, para para iniciarse en el desarrollo de aplicaciones avanzadas y profesionales.

Contenidos:

Clases, atributos y métodos. Variables, operadores, tipos, conversiones de tipos. Estructura general de un programa. Atributos, modificadores de acceso, constructores, sobrecarga y estructura condicional. Estructuras de iteración, condicional múltiple y arreglos. Herencia. Clases abstractas. Clases de interface. Polimorfismo. Aplicaciones sobre listas implementadas en arrays. Diseño de estructuras de datos genéricas: Aplicaciones sobre listas, pilas y colas. Control manual de homogeneidad. Generics, control de homogeneidad mediante parametrización de clases en Java. Implementación del patrón Iterator mediante las interfaces de la API de Java. Gestión de paquetes de clases en Java.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 2.) Weiss, M. A. (2000). "Estructuras de Datos en Java – Compatible con Java 2". Madrid: Addison Wesley. ISBN: 84-7829-035-4 [\[disponible en biblioteca central\]](#)
- 3.) Documentación oficial de la plataforma Java (Javadoc).
- 4.) Deitel, H., Deitel, P. (2005 o posterior). "Java Cómo Programar" . México: Prentice Hall. ISBN: 970-26-0518-0 [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 5.) Drozdek, A. (2007). "Estructura de Datos y Algoritmos en Java". México D.F.: Thomson. ISBN: 9789706866110 [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 6.) Eckel, B. (2002 aunque existe edición posterior). "Piensa en Java". Madrid: Pearson Educación. ISBN: 9788489660342. [\[disponible en biblioteca del Departamento de Sistemas\]](#)

Evaluación:

Evaluación de contenidos y prácticas en cuestionarios 1, 2 y 3, desafíos de programación 1 y 2, trabajo práctico integrador y parcial único.

Unidad N° 2: Excepciones. Archivos de Texto. Serialización. Expresiones Regulares.

Objetivos específicos:

Que el estudiante conozca, domine y aplique técnicas de captura, tratamiento y modelado de errores en tiempo de ejecución, para lograr control profesional de las aplicaciones y componentes que desarrolle.

Que el estudiante conozca, domine y aplique elementos y técnicas de procesamiento de archivos de texto, para usarlos como fuente estándar de almacenamiento para el diseño de pruebas sobre grandes lotes de datos.

Que el estudiante conozca, domine y aplique técnicas de serialización, para resguardar y recuperar el estado de un objeto en los momentos que se requiera.

Que el estudiante conozca, domine y aplique elementos básicos de tratamiento de expresiones regulares, para lograr un mayor control en la búsqueda y gestión de patrones de texto cuando le sea requerido.

Contenidos:

Gestión de excepciones en Java 7 y versiones superiores. Jerarquía de clases de excepción. Excepciones chequeadas y no chequeadas. Lanzamiento y captura de excepciones. El bloque try-catch-finally, try with resources y multcatch. Excepciones del programador (excepciones propias).

Manejo del Sistema de Archivos sumando las clases de la API NIO.2 en el paquete java.nio.file, Interfaz Path y clases Files y FileSystem. Serialización. La interface Serializable. Clases de entrada y salida para la persistencia de objetos. Control de excepciones. Aplicaciones de la Serialización para dar persistencia a objetos completos. Introducción al uso de expresiones regulares para manejo de archivos de texto.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Eckel, B. (2002 aunque existe edición posterior). "Piensa en Java". Madrid: Pearson Educación. ISBN: 9788489660342. [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 2.) Weiss, M. A. (2000). "Estructuras de Datos en Java – Compatible con Java 2". Madrid: Addison Wesley. ISBN: 84-7829-035-4 [\[disponible en biblioteca central\]](#)
- 3.) Documentación oficial de la plataforma Java (Javadocs).
- 4.) Deitel, H., Deitel, P. (2005 o posterior). "Java Cómo Programar" . México: Prentice Hall. ISBN: 970-26-0518-0 [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 5.) Drozdek, A. (2007). "Estructura de Datos y Algoritmos en Java". México D.F.: Thomson. ISBN: 9789706866110 [\[disponible en biblioteca del Departamento de Sistemas\]](#)

Evaluación:

Evaluación de contenidos y prácticas en cuestionarios 4 y 5, desafíos de programación desde el 2 en adelante, trabajo práctico integrador y parcial único.

Unidad N° 3: Streams y Expresiones Lambda..

Objetivos específicos:

Que el estudiante conozca, domine y aplique elementos básicos de la API de Streams y de Expresiones Lambda para que pueda producir código conciso y significativo, además de permitir una aproximación a la programación funcional en Java.

Contenidos:

Métodos anónimos. Interfaces Funcionales. Elementos fundamentales de la API de Streams. El package java.util.stream. Operaciones intermedias y operaciones terminales. Iteración externa vs. Iteración interna. Creación de instancias de streams. Operaciones básicas con streams. Expresiones Lambda en Java: funciones como entidades de primer nivel, Paso de comportamientos como parámetro. Clases de interface con métodos implementados por default. Interfaces funcionales. Alcance de expresiones Lambda.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Documentación oficial de la plataforma Java en el sitio de Oracle.

Evaluación:

Evaluación de contenidos y prácticas en cuestionario 6, desafíos de programación desde el 3 en adelante, y en el trabajo práctico integrador.

Unidad N° 4: Análisis y Diseño de Estructuras para Búsqueda y Ordenamiento.

Objetivos específicos:

Que el estudiante incorpore elementos de análisis de algoritmos y las principales formas de notación, para comprender la forma de realizar comparaciones de rendimiento y seleccionar correctamente un algoritmo de acuerdo al contexto de aplicación.

Que el estudiante adquiera dominio de las principales estructuras de datos para búsqueda y/o ordenamiento y sus algoritmos asociados, para aplicación en diversas situaciones reales.

Que el estudiante adquiera conocimiento general de las principales clases nativas de Java para gestión de colecciones, de forma de poder comparar rendimientos entre esas estructuras nativas y las implementaciones obtenidas en el curso.

Contenidos:

Revisión de la Recursividad Algoritmos recursivos y Backtracking. Revisión de la estrategia Divide y Vencerás: Merge Sort y Quick Sort. Revisión de elementos básicos de análisis de algoritmos y la notación Big O. Análisis de eficiencia. Comparación de rendimientos. Relaciones de Recurrencia. Teorema Maestro. Concepto de grupo de ordenamiento (o montículo binario o Heap) y su implementación en un arreglo. Implementación de Heaps. Operaciones de inserción y remoción. Heap Sort. Análisis de eficiencia. Aplicaciones. Árboles binarios y árboles binarios de búsqueda. Árboles equilibrados. Árboles AVL. Inserción y borrado en árboles AVL. Análisis de eficiencia. Revisión de procesos de inserción en árboles AVL: análisis de aspectos del código fuente. Búsqueda por dispersión de claves: Hashing. Diversas técnicas: direccionamiento abierto, encadenamiento separado. Funciones de dispersión. Treaps. Introducción de elementos de aleatoriedad en la generación de un árbol de búsqueda para simplificar el mantenimiento del equilibrio. Combinación entre árboles binarios de búsqueda y montículos binarios. Implementación de Treaps. Operaciones de inserción, equilibrado y remoción. Análisis de eficiencia. Listas de Salto (Skip Lists) como variante para los árboles equilibrados. Implementación de Skip Lists. Algoritmos de inserción, remoción y búsqueda. Análisis de eficiencia. Análisis general de paquete java.util. Clases provistas en java.util para la gestión de estructuras de datos típicas: ArrayList, LinkedList, TreeSet, HashSet, HashTable, Vector, Stack.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Eckel, B. (2002 aunque existe edición posterior). "Piensa en Java". Madrid: Pearson Educación. ISBN: 9788489660342. [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 2.) Weiss, M. A. (2000). "Estructuras de Datos en Java – Compatible con Java 2". Madrid: Addison Wesley. ISBN: 84-7829-035-4 [\[disponible en biblioteca central\]](#)
- 3.) Documentación oficial de la plataforma Java (Javadocs).
- 4.) Deitel, H., Deitel, P. (2005 o posterior). "Java Cómo Programar" . México: Prentice Hall. ISBN: 970-26-0518-0 [\[disponible en biblioteca del Departamento de Sistemas\]](#)
- 5.) Drozdek, A. (2007). "Estructura de Datos y Algoritmos en Java". México D.F.: Thomson. ISBN: 9789706866110 [\[disponible en biblioteca del Departamento de Sistemas\]](#)

Evaluación:

Evaluación de contenidos y prácticas en cuestionario 7, 8 y 9, desafíos de programación desde el 4 en adelante, en el trabajo práctico integrador y el en parcial único.

Unidad N° 5: Gestión de Bases de Datos con Java (JDBC)

Objetivos específicos:

Que el estudiante conozca, domine y aplique los elementos y técnicas disponibles en el marco de JDBC, para posibilitar el diseño de aplicaciones profesionales que accedan a bases de datos.

Contenidos:

JDBC (Java Database Connectivity): definiciones, conceptos y aplicaciones básicas. Análisis de las API de Java para JDBC. Drivers JDBC. Gestión de una conexión a una base de datos. Elementos de SQL. Creación y configuración de una base de datos. Sentencias SQL embebidas en una aplicación Java. Sentencias SQL precompiladas. Transacciones. Operaciones típicas sobre una base de datos: altas, bajas, modificaciones, consultas y sentencias SQL para su desarrollo. Análisis de patrones comunes en el contexto de JDBC. Introducción a JPA y aplicaciones.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Documentación oficial de la plataforma Java (Javadocs).

Evaluación:

Evaluación de contenidos y prácticas en el trabajo práctico integrador.

Unidad N° 6: Interfaces Gráficas de Usuario con JavaFX.

Objetivos específicos:

Que el estudiante conozca, domine y aplique elementos básicos sobre diseño de interfaces visuales en base al framework JavaFX, para posibilitar el diseño general de interfaces de usuario en aplicaciones profesionales.

Contenidos:

Estructura de una aplicación con IGU basada en JavaFx. Creación de formularios en JavaFX. Inclusión de objetos visuales en un formulario JavaFX. Control de eventos. Creación de la IGU de una aplicación con FavaFx. Control de estilo de visualización.

Bibliografía Obligatoria:

- 1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [\[disponible en el aula virtual de la asignatura\]](#)

Bibliografía Complementaria:

- 1.) Documentación oficial de la plataforma Java (Javadocs).

Evaluación:

Evaluación de contenidos y prácticas en el trabajo práctico integrador.

Metodología de enseñanza y aprendizaje

- ✓ Cada clase se realiza en el laboratorio de informática.
- ✓ En cada clase se dicta un tema central, y el mismo se ejemplifica y analiza con modelos de programas presentados por los profesores.
- ✓ Los alumnos realizan modificaciones y variantes sobre los modelos, y realizan ejercicios nuevos en base a los temas tratados.
- ✓ Los trabajos prácticos y las tareas semanales integran los

	<p>conocimientos de las distintas unidades, y parte del tiempo de clase se usa para analizar dudas y elementos relevantes referidos a esos trabajos.</p>																																	
<p>Sistema de evaluación (Nombrar y describir cada una de las diferentes instancias de evaluación)</p>	<ul style="list-style-type: none"> ✓ La materia se evalúa con un parcial único (que se puede recuperar), un trabajo práctico único, y una nota promedio entre diversas actividades prácticas (como cuestionarios y tareas especiales de programación designadas como "desafíos de programación"). ✓ El tema de cada actividad evaluable integra conocimientos de las distintas unidades del programa, y tiene complejidad y volumen de trabajo acorde a la necesidad de evaluar a los estudiantes. 																																	
<p>Regularidad: condiciones (Describir las condiciones necesarias para regularizar. Se sugiere incluir la aclaración que el estudiante en condición de regular puede rendir en el plazo de un ciclo lectivo sin control de correlativas aprobadas)</p>	<p>Todo alumno obtenga al menos una calificación de 4(cuatro) en cada una de las tres evaluaciones generales: el parcial, el práctico integrador y el promedio de actividades prácticas de aula, pero con promedio de 6(seis) o más entre las tres, obtendrá la aprobación directa: no deberá rendir examen final. Simplemente deberá inscribirse en cualquier turno en que tenga ocasión de hacerlo, y pasar el día del examen a hacer firmar la libreta con su nota final, que en todos los casos será igual al promedio simple y redondeado entre las tres notas de regularidad.</p> <p>Si el promedio entre esas tres calificaciones fuese menor que 6(seis), el alumno deberá hacer un coloquio de recuperación. En caso de aprobar este coloquio con nota 6(seis) o más, el alumno obtendrá la condición de aprobación directa y su nota final será igual a 6(seis). Pero si el alumno obtiene nota menor a 6(seis) en este coloquio, quedará libre.</p> <p>También quedará libre el alumno que no llegue al 4(cuatro) en una o más de las tres evaluaciones, aunque podrá recuperar el parcial único (y sólo el parcial único) en caso de no aprobarlo en la fecha original.</p> <p>A partir de 2017, todo alumno puede inscribirse a rendir una asignatura dentro de un período de ciclo lectivo completo a partir del momento de la regularización, sin que afecte el cumplimiento de las correlatividades previas. A partir de vencido ese plazo de un ciclo completo, para poder inscribirse a examen deberá cumplir primero con las correlatividades previas.</p> <p>A los fines estadísticos, se considerará que un alumno abandonó el cursado si sólo hizo o desarrolló una de las tres instancias de evaluación previstas, importar si la aprobó o no.</p> <p>Escala de notas aplicable DURANTE el cursado(*)</p> <table border="1" data-bbox="678 1482 1218 1871"> <thead> <tr> <th>NOTAS</th> <th>PORCENTAJE</th> <th>CALIFICACIÓN</th> </tr> </thead> <tbody> <tr> <td>1</td> <td></td> <td>No Aprobado</td> </tr> <tr> <td>2</td> <td></td> <td>No Aprobado</td> </tr> <tr> <td>3</td> <td></td> <td>No Aprobado</td> </tr> <tr> <td>4</td> <td>55% a 57%</td> <td>Aprobado</td> </tr> <tr> <td>5</td> <td>58% a 59%</td> <td>Aprobado</td> </tr> <tr> <td>6</td> <td>60% a 68%</td> <td>Aprobado</td> </tr> <tr> <td>7</td> <td>69% a 77%</td> <td>Aprobado</td> </tr> <tr> <td>8</td> <td>78% a 86%</td> <td>Aprobado</td> </tr> <tr> <td>9</td> <td>87% a 95%</td> <td>Aprobado</td> </tr> <tr> <td>10</td> <td>96% a 100%</td> <td>Aprobado</td> </tr> </tbody> </table> <p>(*) Escala acordada en reunión de Docentes Coordinadores</p>	NOTAS	PORCENTAJE	CALIFICACIÓN	1		No Aprobado	2		No Aprobado	3		No Aprobado	4	55% a 57%	Aprobado	5	58% a 59%	Aprobado	6	60% a 68%	Aprobado	7	69% a 77%	Aprobado	8	78% a 86%	Aprobado	9	87% a 95%	Aprobado	10	96% a 100%	Aprobado
NOTAS	PORCENTAJE	CALIFICACIÓN																																
1		No Aprobado																																
2		No Aprobado																																
3		No Aprobado																																
4	55% a 57%	Aprobado																																
5	58% a 59%	Aprobado																																
6	60% a 68%	Aprobado																																
7	69% a 77%	Aprobado																																
8	78% a 86%	Aprobado																																
9	87% a 95%	Aprobado																																
10	96% a 100%	Aprobado																																

Promoción: condiciones	No aplica.																																	
<p>Aprobación Directa: condiciones. (la calificación será la nota registrada como Nota Final en Autogestión) (Se sugiere incluir la aclaración que el estudiante, en esta condición, puede registrar su nota en examen en el plazo de un ciclo lectivo, sin control de correlativas aprobadas, y después de ello se le exigirán correlativas aprobadas)</p>	<p>Los alumnos que hayan obtenido la aprobación directa no deberán rendir examen final. Simplemente deberán inscribirse en cualquier turno en que tengan ocasión de hacerlo, y pasar el día del examen a hacer firmar la libreta con la nota final, que en todos los casos será igual al promedio simple y redondeado entre las tres notas de regularidad., o directamente 6(seis) si el alumno llegó a esta condición a través del coloquio de recuperación. La condición de aprobación directa no tiene vencimiento.</p> <p>A partir de 2017, todo alumno puede inscribirse a rendir una asignatura dentro de un período de ciclo lectivo completo a partir del momento de la regularización, sin que afecte el cumplimiento de las correlatividades previas. A partir de vencido ese plazo de un ciclo completo, para poder inscribirse a examen deberá cumplir primero con las correlatividades previas.</p>																																	
<p>Modalidad de examen final (Describir las características metodológicas del examen final para los distintos estados del estudiante)</p>	<p>Como se indicó, no hay examen final: los alumnos que regularicen tienen aprobada la materia con nota final igual al promedio entre sus notas de regularidad.</p> <p>Escala de Notas para Examen Final (*)</p> <table border="1" data-bbox="771 892 1307 1281"> <thead> <tr> <th>NOTA</th> <th>PORCENTAJE</th> <th>CALIFICACIÓN</th> </tr> </thead> <tbody> <tr><td>1</td><td></td><td>Insuficiente</td></tr> <tr><td>2</td><td></td><td>Insuficiente</td></tr> <tr><td>3</td><td></td><td>Insuficiente</td></tr> <tr><td>4</td><td></td><td>Insuficiente</td></tr> <tr><td>5</td><td></td><td>Insuficiente</td></tr> <tr><td>6</td><td>60% a 68%</td><td>Aprobado</td></tr> <tr><td>7</td><td>69% a 77%</td><td>Bueno</td></tr> <tr><td>8</td><td>78% a 86%</td><td>Muy Bueno</td></tr> <tr><td>9</td><td>87% a 95%</td><td>Distinguido</td></tr> <tr><td>10</td><td>96% a 100%</td><td>Sobresaliente</td></tr> </tbody> </table> <p>(*) Escala acordada en reunión de Docentes Coordinadores</p>	NOTA	PORCENTAJE	CALIFICACIÓN	1		Insuficiente	2		Insuficiente	3		Insuficiente	4		Insuficiente	5		Insuficiente	6	60% a 68%	Aprobado	7	69% a 77%	Bueno	8	78% a 86%	Muy Bueno	9	87% a 95%	Distinguido	10	96% a 100%	Sobresaliente
NOTA	PORCENTAJE	CALIFICACIÓN																																
1		Insuficiente																																
2		Insuficiente																																
3		Insuficiente																																
4		Insuficiente																																
5		Insuficiente																																
6	60% a 68%	Aprobado																																
7	69% a 77%	Bueno																																
8	78% a 86%	Muy Bueno																																
9	87% a 95%	Distinguido																																
10	96% a 100%	Sobresaliente																																
<p>Actividades en laboratorio</p>	<ul style="list-style-type: none"> ✓ Todas las clases de la materia se desarrollan en el laboratorio de informática. ✓ Cada alumno debe instalar en su computadora personal las herramientas de software requeridas por la materia, y realizar en forma personal prácticas y pruebas sobre ellas para lograr dominio pleno de su uso. ✓ Es bienvenida la utilización por parte de los alumnos de sus propias notebooks, en el transcurso de cada clase. ✓ En varias clases se prevé la realización de trabajos y tareas pautadas para ser teminadas (en la medida de lo posible) en el transcurso de la misma clase. Esos trabajos se suben al aula virtual para su revisión por parte de los jefes de trabajos prácticos y auxiliares del curso. 																																	
Horas/año totales de la asignatura (hs. cátedra)	128 horas cátedra en total, todas en el segundo cuatrimestre (16 semanas de clase, 8 horas cátedra por semana).																																	
Cantidad de horas	64 horas.																																	

prácticas totales (hs. cátedra)	
Cantidad de horas teóricas totales (hs. cátedra)	64 horas.
Tipo de formación práctica (sólo si es asignatura curricular -no electiva-)	No aplica.
Cantidad de horas cátedras afectadas a la formación práctica indicada en el punto anterior (sólo si es asignatura curricular -no electiva-)	No aplica.
Descripción de los prácticos	<ul style="list-style-type: none"> ✓ Desarrollo de programas basados en diversas estructuras de datos estudiadas y en diversas estrategias de resolución de problemas. ✓ Desarrollo de programas que apliquen técnicas de búsqueda basadas en estructuras de búsqueda rápida. ✓ Desarrollo programas que apliquen tecnologías Java para acceso a bases de datos con JPA y con interfaces visuales de usuario basadas en JavaFx.
Criterios generales (los cuales serán tenidos en cuenta en las correcciones)	<p>Los trabajos se entregan a través del aula virtual, y son revisados por los docentes de la cátedra. También se plantean cuestionarios y desafíos de corrección automática. El enunciado de cada trabajo incluye consignas a cumplir y recomendaciones adicionales, además de criterios de evaluación tales como: compilación, completitud de consignas cubiertas, diseño de clases, lógica aplicada en la solución de cada ítem, diseño de interfaz de usuario y elementos de control de carga e integridad. Cada criterio se evalúa y aporta un peso a la calificación final. La escala de conversión de notas es la que se indicó más arriba en secciones previas de este mismo documento.</p> <p>Los alumnos deben presentar a través del aula virtual proyectos desarrollado en plataforma Java, que incluya los archivos fuente de cada clase implementada, más los paquetes externos que pudieran haber usado. Cuando se trate de cuestionarios y desafíos de programación, los alumnos simplemente deben subir sus respuestas mediante los recursos disponibles en el aula virtual.</p>
Cronograma de actividades de la asignatura (contemplando las fechas del calendario 2017 y para cada unidad)	Ver archivo CRTSB2017.doc anexo a esta presentación.
Propuesta para la atención de consultas y mail de contacto.	<p>Email de los docentes:</p> <p>Ing. Valerio Frittelli: vfrittelli@gmail.com Ing. Gustavo Federico Bett: gfbett@gmail.com Ing. Felipe Steffolani: fsteffolani@gmail.com Ing. Diego Serrano: diegojserrano@gmail.com Ing. Romina Teicher: rteicher@gmail.com Ing. Marcela Tartabini: mtartabini@gmail.com</p> <p>En caso de requerir horarios de consulta, serán pactados en el momento con los alumnos que lo requieran. TODO el trabajo de la cátedra está permanentemente disponible para los alumnos en el espacio virtual de la cátedra (http://uv.frc.utn.edu.ar) Desde esa aula virtual se canalizan todas las comunicaciones, trabajos prácticos, subida y bajada de archivos, apuntes, notas de clases, foros, planillas de calificaciones, etc.</p>

Plan de integración con otras asignaturas	<p>La asignatura se basa en dos previas: Algoritmos y Estructuras de Datos (de primer año) y Paradigmas de Programación (de segundo año). De ambas, el alumno debe traer un adecuado dominio de la programación de objetos y fundamentos de Java, además de conocimientos sólidos de las estructuras de datos tocadas en esas asignaturas.</p> <p>La materia es la base para el posterior cursado de Diseño de Lenguajes de Consulta (Electiva Cuarto Nivel).</p>																								
Bibliografía Obligatoria	<p>1.) Frittelli, Steffolani, Bett, Serrano, Teicher, Tartabini (2017). "Fichas de Clase para TSB". Córdoba: UTN FRC [disponible en el aula virtual de la asignatura]</p>																								
Bibliografía Complementaria	<p>1.) Eckel, B. (2002 aunque existe edición posterior). "Piensa en Java". Madrid: Pearson Educación. ISBN: 9788489660342. [disponible en biblioteca del Departamento de Sistemas]</p> <p>2.) Weiss, M. A. (2000). "Estructuras de Datos en Java – Compatible con Java 2". Madrid: Addison Wesley. ISBN: 84-7829-035-4 [disponible en biblioteca central]</p> <p>3.) Documentación oficial de la plataforma Java (Javadocs).</p> <p>4.) Deitel, H., Deitel, P. (2005 o posterior). "Java Cómo Programar" . México: Prentice Hall. ISBN: 970-26-0518-0 [disponible en biblioteca del Departamento de Sistemas]</p> <p>5.) Drozdek, A. (2007). "Estructura de Datos y Algoritmos en Java". México D.F.: Thomson. ISBN: 9789706866110 [disponible en biblioteca del Departamento de Sistemas]</p>																								
Distribución de docentes	<table border="1"> <thead> <tr> <th><i>Curso</i></th> <th><i>Turno</i></th> <th><i>Día y Horas</i></th> <th><i>Profesor</i></th> <th><i>JefeTrab.Práct.</i></th> <th><i>Ayudante</i></th> </tr> </thead> <tbody> <tr> <td>3K1</td> <td>mañana</td> <td>Mie 1-2-3-4 Jue 4-5-6-7</td> <td>Frittelli</td> <td></td> <td>Serrano</td> </tr> <tr> <td>3K3</td> <td>tarde</td> <td>Mar 3-4-5-6 Vie 3-4-5-6</td> <td>Steffolani</td> <td></td> <td>Teicher</td> </tr> <tr> <td>3K4</td> <td>noche</td> <td>Mar 3-4-5-6 Mie 3-4-5-6</td> <td>Bett</td> <td></td> <td>Tartabini</td> </tr> </tbody> </table>	<i>Curso</i>	<i>Turno</i>	<i>Día y Horas</i>	<i>Profesor</i>	<i>JefeTrab.Práct.</i>	<i>Ayudante</i>	3K1	mañana	Mie 1-2-3-4 Jue 4-5-6-7	Frittelli		Serrano	3K3	tarde	Mar 3-4-5-6 Vie 3-4-5-6	Steffolani		Teicher	3K4	noche	Mar 3-4-5-6 Mie 3-4-5-6	Bett		Tartabini
<i>Curso</i>	<i>Turno</i>	<i>Día y Horas</i>	<i>Profesor</i>	<i>JefeTrab.Práct.</i>	<i>Ayudante</i>																				
3K1	mañana	Mie 1-2-3-4 Jue 4-5-6-7	Frittelli		Serrano																				
3K3	tarde	Mar 3-4-5-6 Vie 3-4-5-6	Steffolani		Teicher																				
3K4	noche	Mar 3-4-5-6 Mie 3-4-5-6	Bett		Tartabini																				

Firma:

Aclaración: