

## **Escuela Superior de Comercio Nº 43**

Carrera: Técnico Superior en Desarrollo de Software

Unidad Curricular: Ingeniería del Software II

Horas semanales: 4 (cuatro)

Curso: 2º año. División: Única.

Profesor: Silvio Héctor Moschen.

Fundamentación: Diseñar Artefactos de Software que resuelvan problemas. Adquirir los conocimientos para llevar a la práctica Aplicaciones Multicapas abordadas desde el Paradigma Orientado a Objetos y reuso de Componentes.

Aplicar Patrones de Diseño como Model View Controller, Unit of Work, IoC que satisfagan las necesidades para aplicaciones Rest, Mobile, Web, entre otras.

Objetivo: El objetivo es que el alumno aprenda a abordar el desarrollo de software respetando patrones de diseño, ajustándose a frameworks y a las nuevas tecnologías del mercado, en una forma de trabajo grupal y colaborativa.

- Adquirir los conocimientos del Paradigma de Objetos.
- Aprender a unificar las etapas de análisis, diseño y programación de sistemas.
- Separar una Aplicación en Capas.
- Resolver aspectos vinculados a la arquitectura de un proyecto de software.
- Familiarizarse con los Frameworks de la Industria del Software.
- Integrar diferentes tecnologías a los efectos de mejorar la Productividad.
- Aplicar los conocimientos en Proyectos de Software a partir de Casos de Estudio propuestos.

### Organización de los contenidos conceptuales

Unidad 1: Sistemas. Conceptos generales. Sistemas y Sub Sistemas. Evolución en los Sistemas. Impacto de la tecnología. Sistemas informáticos. Análisis, diseño y programación. Principios del diseño: descomposición, desacoplamiento, cohesión, reusabilidad, portabilidad, testeabilidad, flexibilidad, escalabilidad. Arquitecturas distribuidas.

Unidad 2: Diseño Orientado a Objetos. Diseño estructurado. Diseño Orientado al reuso de componentes. Modelo de Objetos. Clases. Objetos. Constructores y Destruyores. Métodos Accesores. Encapsulamiento. Asociaciones. Cardinalidades. Herencia de Clases. Polimorfismo. Interfaces. Implementación de interfaces. Colecciones de Objetos. Listas. Arrays. Agregaciones y Composiciones. Diagramas de Clases.

Unidad 3: Persistencia de Objetos. La necesidad de persistir. Carencias nativas de persistencia. Alternativas al manejo de persistencia. Serialización. Bases de Datos. Mapeo Objeto-Relacional. Implementación del mapeo Objeto-Relacional. Frameworks de persistencia. Aplicación de frameworks de persistencia. Test unitarios.

Unidad 4: Abstracción de proyectos en Capas. Entidades (Modelo). Acceso a Datos (DAO). Servicios (Services). Patrones de diseño a utilizar: IoC (Inject of Dependency), Unit of Works, Interceptors, AOP (Aspect Oriented Programming). Implementación de capas en proyectos de software. La organización para el trabajo en equipo.

Unidad 5: Arquitectura MVC (Model – View - Controller). Controladores como servicios de orquestación en aplicaciones. Los modelos. Las Vistas. Interfaces de Usuario. Validación de Datos. Prototipos rápidos. Frameworks MVC. API Rest. Conceptos. Implementaciones. Serialización de Objetos. Integración con otras aplicaciones.

Unidad 6: Prueba unitaria. Integración. Validación y pruebas de un proyecto. Refactorización del código. Diseño de artefactos que no componen el código: documentación, archivos de ayuda, capacitaciones. Auditoria y pruebas: escalabilidad, rendimiento, balanceo de cargas, estrategias de caching, etc.

#### Organización de los contenidos procedimentales

- Planteamiento de preguntas y problemas.
- Resolución de casos prácticos de estudio propuestos por el docente.
- Selección de Frameworks que resuelvan la arquitectura planteada. Integración entre diferentes Frameworks. Configuraciones.
- Formulación de soluciones a casos propuestos.
- Seguimiento individual de prácticas en máquina.

#### Organización de los contenidos actitudinales

- Respeto por las ideas y opiniones producidas por los otros.
- Cooperación del grupo para resolver casos complejos.
- Defensa de las propias opiniones, ideas y reflexiones.
- Valoración del intercambio de ideas como elemento constructivo y de enriquecimiento personal.
- Valoración de la investigación como fuente de conocimiento y aprendizaje.

#### Estrategias Metodológicas

- Exposición, intercambio y debate abierto. Prácticas en máquina.

#### Modalidades de Cursado

- Regular con cursado presencial
- Libre
- Semi presencial

## Evaluación

### Instrumentos de Evaluación

A fin de permitir al docente un panorama más ajustado acerca el estado del saber de cada uno de los alumnos, se considera necesario utilizar en forma complementaria y no excluyente instrumentos tales como:

- Trabajos prácticos con carácter evaluador.
- Observación directa – por parte del docente- del trabajo individual.
- Exámenes Parciales.

### Criterios de evaluación:

- Validaciones propuestas y generalización de soluciones.
- Cumplimiento de las tareas solicitadas.
- Respeto por el trabajo y las opiniones del resto de los compañeros.
- Resolución de los Casos Propuestos.
- Desarrollo de un proyecto individual (tema propuesto por el alumno)

### Para regularizar en forma presencial, los alumnos deberán:

- Tener un mínimo de 75% de asistencia a clases durante el tiempo que dure la cursada. Para el alumno que presente certificado de trabajo y/o se encuentre en otras situaciones excepcionales, debidamente comprobadas, el mínimo es de 50% de asistencia. Las asistencias se computan en forma cuatrimestral.
- Aprobar el 80% de los trabajos prácticos, entregados en tiempo y forma  
Aprobar un examen parcial en la primera instancia o su recuperatorio, con una nota mínima de 6(seis).

### Para regularizar en forma semi-presencial, los alumnos deberán:

- Tener un mínimo de 40% de asistencia a clases en cada cuatrimestre.
- Aprobar el 100% de los trabajos prácticos
- Aprobar un examen parcial en la primera instancia o su recuperatorio, con una nota mínima de 6 (seis).

### Para acceder a la Promoción Directa los alumnos deberán:

- Cumplir con el porcentaje de asistencia establecido para el régimen presencial.
- Aprobar el 100% de los trabajos prácticos, entregados en tiempo y forma
- Aprobar un examen parcial, en primera instancia, con un mínimo de 8 (ocho)
- Aprobar una instancia final integradora, con un mínimo de 8 (ocho), este examen no tiene recuperatorio.

### Observaciones:

- Si el alumno regulariza Informática II pero no la promociona, tiene derecho a un examen final escrito, en cualquiera de los turnos correspondientes, debiendo obtener una calificación mínima de 6 (seis). Mantiene la regularidad durante 3 (tres) años consecutivos a partir del primer turno correspondiente al año lectivo siguiente al de la cursada

### Acreditación para el alumno libre:

El estudiante libre deberá aprobar un examen final, con modalidad escrita, obteniendo una calificación mínima de 6 (seis)

### Cronograma

- Primera parte: **Adquirir los conceptos teóricos** a desarrollar durante el ciclo lectivo (unidades 1, 2).
- Segunda parte: **Implementar casos de estudio de forma Práctica**, continuar con el aprendizaje de los conceptos teóricos llevándolos a la práctica a través de software y herramientas apropiadas (unidades 3, 4, 5 y 6).

### Bibliografía:

Rumbaugh J., “Modelado y Diseño Orientado a Objetos”  
Editorial Prentice Hall – 1997.

Rumbaugh J., Jacobson I., Booch G.  
“El Lenguaje Unificado de Modelado. Manual de Referencia”

Editorial Addison-Wesley – 2000.

Larman C., “UML y Patrones” Segunda Edición – Editorial Prentice-Hall – 2003.

Odstreil M., “Apuntes de Conceptos de Sistemas y Análisis Orientado a Objetos”  
Primera Edición – FACET, UNT – 2001.

Yourdon, E., “Análisis Estructurado Moderno”  
Primera Edición – Editorial Prentice Hall – 1993

Pfleeger S., “Ingeniería de Software, Teoría y Práctica”  
Primera Edición – Editorial Prentice Hall – 2002.

Gómez A., Juristo N., Montes C., Pazos J.  
“Ingeniería del Conocimiento” – Primera Edición  
Editorial Centro de Estudios Ramón Areces, S.A. – 1997

Ediciones y publicaciones de Internet, tutoriales, libros.

Apuntes y Tutoriales de Internet.