

**INSTITUTO TECNOLÓGICO DE LAS AMÉRICAS**  
**CARRERA DE TECNÓLOGO EN DESARROLLO DE SOFTWARE**  
**PROGRAMACIÓN II**

---

**Nombre de la asignatura:** Programación II

**Nomenclatura del Curso:** SOF-004

**Prerrequisitos:** Programación I

**Nomenclatura del prerrequisito:** SOF-003

**Número de Créditos:** 4

**Horas Teóricas:** 30

**Horas de Práctica:** 30

**Horas Investigación:** 45

**Introducción:** Posterior a los conocimientos necesarios para construir una aplicación de uso comercial, impartido en programación I, aprenderá el estudiante en este módulo a trabajar de manera más profesional y crear aplicaciones en tres capas, lo cual le ayudará a dividir la complejidad de los sistemas, y por tanto será un eje de apoyo para crear aplicaciones mucho más complejas que las trabajadas en programación.

**Justificación:** Los sistemas de información pueden crecer enormemente en complejidad, la manera en como se maneje esta, se determinará las posibilidades de éxito del producto final. La división de la complejidad del desarrollo de aplicaciones en tres capas, es el modelo más aceptado hoy día para lidiar la complejidad.

**Descripción:** Esta asignatura abarca metodologías empleadas para segmentar la complejidad de los sistemas informáticos, para fines de hacer las aplicaciones más sostenibles, de más facilidad de modificación y la escalabilidad que los sistemas desarrollados en grandes escalas necesitan.

Los conocimientos adquiridos por el estudiante hasta este nivel le servirán de base para comprender esta nueva perspectiva de visualizar el proceso de desarrollo de software.

**Objetivo General:** El estudiante conocerá maneras efectivas de dividir la complejidad de los sistemas, de forma que sean más simple su escritura, posteriormente más fácil mantenimiento y con alto grado de escalabilidad que le permita crecer conforme

varíen los requerimientos y las tecnologías.

**Objetivos Específicos:**

1. Crear la capacidad de segmentar la complejidad y trabajar sobre problemas complejos.
2. Comprender criterios para ignorar datos irrelevantes para el sistema.
3. Conocer metodologías actualizadas para la concepción de problemas de software

**Contenidos:**

1. Introducción.
  - 1.1. Conceptos fundamentales de la programación multicapas.
  - 1.2. Modelo convencional frente al modelo multicapas
  - 1.3. Escenarios de uso para el modelo multicapas.
2. Capa de Presentación.
  - 2.1. Componentes
  - 2.2. Contenedores
  - 2.3. Separación del modelo y la vista del sistema
  - 2.4. Esquema de organización visual (Layouts)
  - 2.5. Optimización de arquitecturas de presentación
3. Capa de acceso a Datos
  - 3.1. Enfoques de persistencia de los objetos
  - 3.2. Manejos de flujos de datos
  - 3.3. Separación del acceso a bases de datos
4. Capa de lógica de negocios
  - 4.1. Conceptos
  - 4.2. Responsabilidades
  - 4.3. Implementación
  - 4.4. Control de acceso
  - 4.5. Publicación
5. Patrones de Diseño
  - 5.1. Definición
  - 5.2. Patrón de Observador (Observer Pattern)
  - 5.3. Patrón Decorador (Decorator Pattern)
  - 5.4. Patrón de Factoría (Factory Pattern)
  - 5.5. Patrón de instancia única (Singleton Pattern)
  - 5.6. Patrón de comando (Command Pattern)
  - 5.7. Patrón de Adaptador (Adapter Pattern)
  - 5.8. Patrón de Fachada (Facade Pattern)
  - 5.9. Patrón de plantilla de métodos (Template Method Pattern)
  - 5.10. Patrón de iteración (Iterator Pattern)
  - 5.11. Patrón de estado (State Pattern)
  - 5.12. Patrón de Proxy (Proxy Pattern)
6. Esquema Modelo /Vista / Control (MVC).
  - 6.1. Definición

- 6.2. Modelo
- 6.3. Vista
- 6.4. Control

**Metodología:**

Apuntes de clases y guías de los contenidos. El profesor será el facilitador del aprendizaje, dirigiendo los equipos para la investigaciones bibliografías, exposiciones, videos, diálogos, análisis de casos y debates.

**Recursos**

- El estudiante tendrá como recursos los laboratorios de nuestra institución que en el momento se necesiten, previendo que estén disponibles en el momento.
- Recursos bibliográficos en nuestra biblioteca.

**Evaluación:**

Primer parcial	20%
Segundo parcial	20%
Pruebines, talleres y prácticas	30%
Examen general	30%

**Textos:**

Microsof E-learning

**Beginning Java Objects: From Concepts To Code**

(2da edición) 2005

**Autor:** Jaquie Barker

**Editorial :** Apress

ISBN: 1-59059-457-6

**Head First Design Patterns**

Editorial O'Reilly

**Autor:** Katty Sierra

Primera edición: Octubre 2004

ISBN 10: 0-596-00712-4

ISBN 13: 9780596007126

**Profesor :**

Stalin Rodríguez