

Programa**CARRERA: Analista Universitario De Sistemas De Información****PLAN:2014****UNIDAD CURRICULAR: Programación Aplicada I****CURSO: 2 - B****REGIMEN: Anual****ASIGNACION HORARIA SEMANAL:6****PROFESOR/ES: Borrás, Javier Esteban****AÑO: 2019****FUNDAMENTACIÓN/ENCUADRE/PRESENTACIÓN:**

El desafío tecnológico de la última década se basa en la masividad, interconexión y cambio de hábitos sociales y culturales de la mano de internet que creció de una manera desregulada y exponencial, llevó la proliferación de sistemas distribuidos, multiplataforma hasta extender su uso a todas las "cosas" [1].

El continuo desarrollo y la masificación del uso de sistemas informáticos multiplataforma nos ha llevado a convivir con un "ecosistema" en donde habitan distintos lenguajes, patrones de diseño y arquitectura, persistencia de datos y tecnologías. El desarrollador de software tiene que estar preparado para poder plasmar el diseño de algoritmos de la manera más agnóstica[2] posible, y que la elección de un lenguaje de programación en particular sea simplemente aplicar una sintaxis determinada para resolver un planteo abstracto.

La POO (Programación Orientada a Objetos) ha sido el paradigma de desarrollo adoptado por gran parte (sino la mayoría) de la industria del software que permitió el acompañamiento a las necesidades de cambio constantes que nos plantea el mercado y es de alguna manera lo que amalgama todo el ecosistema de lenguajes, paradigmas, modelos de datos y arquitecturas.

Un Analista de Sistemas debe entender cómo a partir de un problema concreto de la vida real, puede llevarlo a un modelo informático abstracto para finalmente interactuar con un usuario final u otro sistema. Por ello, para poder brindar las herramientas de la "transformación" desde lo abstracto un algoritmo hay que conocer las capacidades de los diversos lenguajes y paradigmas de desarrollo para poder identificar y seleccionar la solución sistémica más eficiente y al fin poder adaptarse rápidamente a los requerimientos que sean necesarios para poder emprender un proyecto de software exitoso. El desafío más grande a cumplir en esta materia es poder incentivar al alumno a abrazar el cambio, a mantener el contenido actualizado año a año (e inclusive manteniendo el lineamiento formal de la materia durante el mismo cursado de la misma)

-

OBJETIVOS

- a. Comprender los fundamentos de la POO.
- b. Seleccionar y Aplicar algoritmos y patrones de diseño para resolver problemas de forma eficiente.
- c. Conocer las tendencias de uso de sistemas y los lenguajes de programación más utilizados.
- d. Utilizar de un Entorno Integrado de Desarrollo.
- e. Aprender a utilizar motores de búsqueda en Internet de manera eficiente como herramienta de consulta, colaboración y ampliación de conocimientos.

-

CONTENIDOS

Para lograr los objetivos antes planteados desarrollaremos estos tres ejes durante el transcurso de la materia:

- Programación Aplicada – Algoritmos y estructura de datos
- Fundamentos del lenguaje de programación orientado a objetos aplicados, principal pero no excluyente, en tecnología .Net con el lenguaje de programación denominado C#
- Introducción a aplicaciones Web (modelo cliente-servidor)

PROGRAMA

o **Algoritmos y estructura de datos**

o Algoritmos

§ Definición

- ¿Qué es un algoritmo?
- Precondición
- Postcondición
- Invariantes

o Diseño e implementación

§ Diseño

§ Implementación

§ Algoritmos de Computación

o Estructura de datos

§ Definición

§ Tipos de datos

- Representación de datos en memoria

§ Variables

- Variables Locales
- Variables Instancia
- Variables globales
- Variables estáticas
- Constantes

o **Lenguajes de Programación**

§ Definición

- Función y objetivo

§ Tipos de Lenguajes según su nivel

- Clasificación de los lenguajes de programación

§ Paradigmas de programación

- Paradigmas de programación más usados

o Elementos de un lenguaje de Programación

§ Sintaxis

§ Palabras Reservadas

§ Class Body

§ Typing

§ Open Source Vs Closed Source

§ Compilación

§ Empaquetado

o Microsoft .Net Framework

o Un poco de historia

o Microsoft Visual Studio

o Distribuciones

o Bienvenidos a C#

- o Estructura del programa
- § Tipos y variables
- § Expresiones
- § Declaraciones
- § Clases y objetos
- § Estructuras
- § Arrays
- § Interfaces
- § Enums
- § Delegados
- § Atributos
- o Tipos de Ensamblados
- § Class Libraries
- § Application
- § Web MVC - SPA - REST
- § Xamarin
- § Windows Presentation Foundation
- § Windows Communication Foundation
- § Console Application
- o **Fundamentos del lenguaje de programación orientado a objetos**
- o Introducción a la POO
- o Conceptos fundamentales
- § Clases
- § Diagramas básicos de representación de clases (UML)
- § Herencia
- § Evento
- § Atributos
- § Mensaje
- § Estado interno
- o Funciones y Métodos
- o Objetos
- o Características de la POO
- § Herencia
- § Abstracción
- § Encapsulamiento
- § Polimorfismo
- § Modularidad
- § Principio de ocultación
- o Interface
- § Sobrecarga de métodos
- § Constructores, Destructores
- § DTO (Data Transfer Object) Concepto

- o **Introducción a la programación web con MVC**
- o MVC: Model View Controller: Patrón de diseño
- o Model: Representación de modelos de datos
- o Views: Concepto, introducción al lenguaje Razor
- o Controller: Métodos, Acciones, Verbos
- o Introducción a Nuget Packages
- o Introducción a Entity Framework

-
-

METODOLOGÍA DE ENSEÑANZA Y RECURSOS A UTILIZAR

La modalidad de trabajo es Teórico-Práctica con ejercicios a desarrollar en aula con computadoras y ejercicios a desarrollar fuera del ámbito áulico.

Se realizarán trabajos prácticos grupales de investigación y desarrollo.

Se utilizarán, presentaciones en power point, videos, además de las prácticas en computadora.

- o Requerimientos de Software:
- o Sistema Operativo Windows Seven o superior
- o Microsoft Visual Studio Community Edition
- o MS SQL Express 2012
- o GitHub (deseable)
- o Microsoft .net 4.5 o superior

EVALUACIÓN

A. CRITERIOS Y SISTEMAS DE EVALUACIÓN

- Partimos desde los conocimientos previos adquiridos por el alumno:
 - o Reconocer y utilizar Diagramas de flujos
 - o Componentes lógicos
 - o Componentes de control
 - o Reconocimiento de lenguaje natural, donde un programa es capaz de comprender (con limitaciones) la información contenida en una expresión lingüística humana.
- Se evaluará:
 - o Que se abstraiga un problema de la vida real para llevarlo a un modelo
 - o informático.
 - o Que reconozca y seleccione el algoritmo más eficiente para resolver el problema planteado.
 - o Que sepa buscar, filtrar y seleccionar en internet la herramienta acorde para determinada problemática.
 - o Presentación, contenido y desarrollo de los trabajos prácticos.

B. CONDICIONES PARA OBTENER LA REGULARIDAD Y/O PROMOCIONALIDAD

MODALIDAD DE CURSADO:

Conforme está establecido en el anexo i de la resolución nº 43/15, la modalidad de cursado será presencial con cursos teórico-prácticos cuatrimestrales o anuales, según lo detallado en la currícula de cada año.

1) Las asignaturas se aprobarán con 4 (cuatro) puntos o más, en una escala de 1 (uno) a 10 (diez)

2) Los estudiantes del Nivel de Pregrado podrán cursar sus estudios en condición de Regular, Promocional o Libre.

3) Serán **alumnos regulares** los que, inscriptos como tales o como promocionales en las fechas y condiciones fijadas al efecto, aprueben:

a) En las asignaturas cuatrimestrales, 2 (dos) exámenes parciales escritos, de los cuales, al menos uno, debe ser individual con una calificación no inferior a 4 (cuatro) puntos cada uno y tengan el 60% de asistencia a clases.

b) En las asignaturas anuales, 3 (tres) exámenes parciales escritos, de los cuales, al menos 2 (dos), deben ser individuales con una calificación no inferior a 4 (cuatro) puntos cada uno y tengan el 60% de asistencia a clases.

En caso de no aprobar o haber estado ausente en un examen parcial, los alumnos tendrán derecho a un examen recuperatorio en la última o penúltima semana de cursado sobre los contenidos del parcial en el que hayan sido aplazados o hayan estado ausentes. De no aprobar o no asistir al examen recuperatorio, serán alumnos libres.

Para aprobar las asignaturas, los alumnos en condición de regular deberán rendir UN EXAMEN

FINAL ANTE TRIBUNAL. De no aprobar las asignaturas en 5 (cinco) turnos de exámenes consecutivos (a partir de la obtención de la regularidad) o desapropar 3 (tres) veces el EXAMEN FINAL, el alumno perderá la condición de REGULAR. La no presentación del alumno a examen no debe interpretarse como aplazo, corresponde AUSENTE.

Aquellos alumnos que perdieren la condición de regular podrán recurrar o rendir libre.

4) Serán **alumnos promocionales** los que, inscriptos como tales en las fechas y condiciones fijadas al efecto, aprueben:

a) en las asignaturas cuatrimestrales, 2 (dos) exámenes parciales escritos de los cuales, al menos uno debe ser individual con una calificación mínima de 7 (siete) puntos cada uno, contando con el 80 % de asistencia a clases.

b) en las asignaturas anuales, 3 (tres) exámenes parciales de los cuales, al menos 2 (dos) deben ser individuales con una calificación mínima de 7 (siete) puntos cada uno, contando con el 80 % de asistencia a clases.

En caso de ausencia a uno de los exámenes parciales, los alumnos tendrán derecho a un examen recuperatorio en la última o penúltima semana de cursado sobre los contenidos del parcial en el que hayan estado ausentes. A los fines de la promoción, no podrán recuperarse parciales con calificación inferior a 7 (siete) puntos.

Dadas estas condiciones, los estudiantes podrán acceder a un coloquio integrador final, que se aprobará con 7 (siete) puntos o más. Este coloquio sólo podrá ser rendido por aquellos alumnos que tengan aprobada/s la/s asignatura/s correlativa/s correspondiente/s.

Los alumnos que perdieren la condición de promocional podrán rendir las asignaturas cumpliendo las condiciones de los alumnos regulares o libres, según el caso.

5) Serán **alumnos libres**, además de los anotados como tales, aquéllos que habiendo cursado asignaturas como promocionales o regulares no cumplan con los requisitos para mantener esa condición. Para aprobar una asignatura como libre los alumnos deberán rendir un EXAMEN FINAL ante tribunal, escrito y oral, ambos ELIMINATORIOS. El examen será sobre el programa completo de la asignatura y se aprobará con una calificación no inferior a 4 (cuatro) en cada una de las instancias.

-
-

BIBLIOGRAFÍA Y/O FUENTES DE CONSULTA

Introducción a la Programación en Java - Apuntes Teóricos y prácticos

Prof. Javier Borrás

Material provisto por la imprenta de la ESCMB (2019) Obligatoria

a. Introducción a la Programación en .Net – Apuntes teóricos, Prof. Javier Borrás (elaboración propia provista por el ESCMB)

b. Guía de Trabajos Prácticos, Prof. Javier Borrás (elaboración propia provista por el ESCMB)

c. Documentación oficial de .Net:

<https://docs.microsoft.com/en-us/dotnet/framework/get-started/> (mar 2019). Básica

d. The Art of Computer Programming by Donald Knuth

ISBN 0-201-03801-3 Básica

e. Object-oriented Design By Coad, P; Yourdon, E. (1991).

ISBN 9780136300700. Ampliatoria

f. Programming: Principles and Practice Using C++ by Bjarne Stroustrup.

ISBN 978-0321543721. Ampliatoria

g. .Net Patterns: Architecture, Design and Process, Christian Thilmany

ISBN 0-32-113002-2. Ampliatoria

h. *HTML 5: Entiende el cambio, aproveche su potencial*
ISBN 978-987-1773-79-4 Ampliatoria

i. *Nanyang Technological University:*
<https://www3.ntu.edu.sg/home/ehchua/programming/index.html> (mar 2019) Ampliatoria

j. *Princeton University:*
<http://introcs.cs.princeton.edu/java/home/> (mar 2019) Ampliatoria

-

Lugar y Fecha: Córdoba, 15 de abril de 2020

Prof. Javier E. Borrás

Firma del Docente Responsable

[1] "cosas" debe tomarse de la definición del inglés: "Internet of Things" (internet de las cosas) es un concepto que se refiere a la interconexión digital de objetos cotidianos con internet. Alternativamente, Internet de las cosas es el punto en el tiempo en el que se conectarán a internet más "cosas u objetos" que personas.

[2] "agnóstica" se refiere a que el diseño de un algoritmo debe ser independiente del lenguaje en que se lo implemente. Donald Knuth, The Art Of Computer Programming

Última modificación: Sunday, 19 de April de 2020, 21:20