

CÓDIGO DE ASIGNATURA

2633

ASIGNATURA: Programación Web 3

REFERENTE DE CÁTEDRA: Ing. Mariano Juiz

AÑO: 2020

CARGA HORARIA: 8

OBJETIVOS:

El programa de la asignatura pretende introducir al alumno en la programación orientada a objetos y la programación web.

Los **propósitos fundamentales** de esta asignatura son:

- Introducir al alumno en el desarrollo de aplicaciones web, brindándole la posibilidad de manejar los conceptos avanzados en la programación web y metodologías que ayudan en la comprensión global de un problema y el planteo de la solución, para su futuro desarrollo.
- Que los lenguajes y herramientas se aprendan y apliquen a casos reales, para que el estudiante entienda cuándo y por qué aplicarlas y pueda aprovecharlos mejor.

Objetivos Generales:

- Familiarizar al alumno en los conceptos y términos avanzados del área de programación y las diferentes metodologías de desarrollo existentes.
- Brindar al alumno los conocimientos necesarios para el desarrollo de aplicaciones web avanzadas utilizando las últimas tecnologías disponibles.
- Enseñar las técnicas clásicas de programación orientada a objetos.
- Incentivar al alumno al trabajo en equipo, lo cual lo preparará para una futura participación en proyectos de software donde los límites de tiempo, los recursos tecnológicos, físicos y humanos y las necesidades de los usuarios cumplen un rol fundamental.

Objetivos Específicos:

- Capacitar al alumno en la comprensión de problemas y planteo de soluciones de manera que sea capaz de resolver y organizar una solución.
- Introducir al alumno en la programación estructurada.
- Capacitar al alumno para que comprenda y maneje con fluidez el concepto de tipo de datos, estructuras de datos, estructuras de control, funciones y procedimientos.
- Capacitar al alumno en el lenguaje de programación C# y aplicaciones web Asp.NET. El alumno creará programas diversos y se enfrentará a problemas reales de programación web.

CONTENIDOS MÍNIMOS:

Introducción a .NET. Introducción al Lenguaje C#. Clases y Objetos en C#. Técnicas de POO en C#. Principios de Arquitectura - SOLID. ASP.NET. MVC. Validación de Controles. Pasaje de Datos entre Formularios. ORM. Acceso a Datos con ADO.NET EF. Excepciones. Servicios Web.

Correlatividades:

- Programación Web 2 (2628)
- Diseño de Aplicaciones Web (2629)
- Visualización e Interfaces (2630)

PROGRAMA ANALÍTICO:

Unidad Nº 1. Introducción a la Programación

- Introducción a la programación.
- Conceptos básicos en programación.
- Fases de desarrollo de un programa.
- Metodologías de programación.
- Lenguajes de máquina, lenguajes ensambladores y lenguajes de alto nivel.
- Comprensión global de un problema.
- Programación Estructurada.
- Estructura general de un algoritmo.
- Diseño de Algoritmos.
- Estructuras Algorítmicas. Secuenciales. Condicionales. Cíclicas. Repetitivas.
- UML. Diagramas elementales. Clases, Casos de Uso

Unidad Nº 2. Introducción a .NET

- ¿Qué es .NET?
- Evolución
- Componentes del Framework

Unidad Nº 3. Fundamentos del Lenguaje C#

- Namespaces
- Clases
- Variables
- Visibilidad
- Sentencias
- Operadores
- Enumeraciones
- Constantes
- Tipos de Proyectos .NET

Unidad Nº 4. Introducción al Lenguaje C#

- Primera página web.
- Primera consola.
- Primer aplicación de escritorio.
- Realización de ejercicios varios.

Unidad Nº 5. Introducción a ASP.Net MVC

- Arquitectura Web
- ASP.NET
- Visión Global de Aplicaciones ASP.NET (Web Pages, SPA, MVC, Web Form, Servicios Web)
- Patrón MVC
- ASP.NET MVC
- Controllers, Views, Models
- Enrutamiento
- Ciclo de vida de la aplicación vs ciclo de vida de la solicitud
- Hola Mundo ASP.MVC
- Ejercicios varios Enrutamiento.

Unidad Nº 6. Operaciones CRUD ASP.Net MVC

- Introducción a Razor
- Vistas fuertemente Tipadas
- Métodos HttpPost
- Model Binding Manual y Automático

Unidad Nº 7. Clases y Objetos en C#

- ¿Qué es un paradigma?
- Diferentes paradigmas
- ¿Qué es la Programación Orientada a Objetos?
- ¿Qué es un Clase?
- ¿Qué es un Objeto?
- ¿Qué es una interfaz?
- Encapsulamiento
- Herencia

- Sobrecarga de métodos
- Instancias
- Creación de Clases e instancias para diferenciar conceptos.
- Constructores y Destructores
- Sobrecarga de Constructores

Unidad N° 8. Técnicas avanzadas POO en C#

- Abstracción
- Miembros Estáticos
- Polimorfismo
- Métodos virtuales – Sobreescritura
- Invocaciones
- Creación de clases Abstractas
- Principios SOLID
- Ejercicios varios

Unidad N° 9. Programación en C#

- Arrays
- Generics
- Colecciones (List)
- Diferencias entre los diferentes conceptos
- ¿Qué son las Excepciones?
- Ejemplos de excepciones.
- Ejercicios forzando errores y su correspondiente tratamiento.

Unidad N° 10. Validación y Vistas ASP.NET MVC

- Layouts
- DataAnnotations
- Partial Views.
- Continuación Razor.

Unidad N° 11. Tratamiento de Datos y ViewModels

- ViewBag, ViewData, TempData, Sesi3n
- ViewModels
- ActionResult (redirecttoaction, redirect, file, json, etc)

Unidad N° 12. Acceso a Datos con ADO.NET Entity Framework (EF)

- Introducci3n ADO.NET
- Patr3n ORM.
- Introducci3n a EF
- Enfoques EF
- Operaciones CRUD sobre 1 entidad.
- Mapeo 1 a 1. 1 a muchos. Muchos a Muchos
- Arquitectura MVC, capas y ORM
- Herencia

- LINQ. LINQ To Objects – LINQ To Entities.

Unidad Nº 13. Servicios Web

- ¿Qué son?
- Ventajas y Desventajas
- Evolución
- SOAP
- XML
- Ejercicios de creación y consumo de Servicios Web XML
- REST Web API
- JSON
- Ejercicios de creación y consumo de Servicios Web JSON

Unidad Nº 14. .Net Core

- Introducción Core
- Net Standard
- Diferencias con FWK

BIBLIOGRAFÍA:

BIBLIOGRAFÍA BÁSICA

Autor	Título	Editorial	Año	Edición
David S. Platt	Así es Microsoft .NET ISBN: 84-481-3251-3	McGraw-Hill	2001	

BIBLIOGRAFÍA COMPLEMENTARIA

Autor	Título	Editorial	Año	Edición
Matthew MacDonald	ASP.NET: manual de referencia ISBN: 84-481-3709-4	McGraw-Hill	2002	

METODOLOGÍA DE ENSEÑANZA:

Metodología General de Clases

La metodología de enseñanza se focaliza en clases teóricas y clases prácticas participativas, con gran cantidad de horas en laboratorio, de manera de lograr que el alumno obtenga un conocimiento equilibrado de los componentes teóricos y prácticos de la materia.

Las clases serán dictadas a través de distintos métodos, como explicaciones a través de definiciones, ejemplos, ejercicios, lectura individual dirigida, actividades grupales de análisis, transferencia, validación colectiva y exámenes. Determinados contenidos temáticos serán presentados a los alumnos a través de proyecciones y videos.

Se desarrollarán diferentes prácticos individuales y/o grupales aplicando los contenidos dados en las diferentes unidades temáticas, para poder fijar los conocimientos de forma práctica. Se fomentará al alumno al trabajo en grupo.

El proceso de enseñanza y de aprendizaje de carácter teórico-práctico, permanente e integral, propone a los alumnos la adquisición de conocimientos, el desarrollo de actitudes y la detección de aptitudes, el aumento de la destreza y las habilidades para comprender y encontrar información relevante, y la resolución de las situaciones nuevas que se le presenten, utilizando un enfoque hacia la resolución de problemas.

El alumno debe mostrar al finalizar el curso un nivel mínimo de destreza en los conceptos y las tecnologías específicas asociadas a la materia.

Las diversas actividades teórico-prácticas planteadas favorecen la investigación, el desarrollo, el trabajo en equipo y la fijación de conocimientos.

Considerando que la adaptación a las nuevas tecnologías supone un reto fundamental actual, se le facilitará al alumno la posibilidad y los medios necesarios para que puedan acceder, conocer e investigar todos los instrumentos que las nuevas y últimas tecnologías ofrecen.

Metodología de Clases Teóricas

- Las clases teóricas están orientadas a introducir a los alumnos en los diferentes conceptos teóricos conceptuales de la materia.
- Cada tema teórico es abordado en clase brindando el profesor ejemplos de aplicación.
- La metodología de trabajo alternará entre clases expositivas donde los profesores explicarán los temas y otras haciendo participar a los alumnos mediante exposición dialogada.

Metodología de Clases Prácticas

- En las clases prácticas los alumnos podrán ejecutar ejercicios junto a los docentes, aplicar los conceptos teóricos, evacuar dudas y aclarar los conceptos necesarios.
- Los alumnos resolverán ejercicios planteados mediante trabajos en grupos o de forma individual, mientras los profesores supervisarán su realización y atenderán consultas personales.
- Las prácticas se referirán a cada núcleo temático de la materia para que el alumno tenga claro qué conceptos está ejercitando. Aquellos ejercicios donde se haga

hincapié en algún concepto fundamental, deben ser supervisados por los profesores en clase, los cuáles harán una conclusión general al final de la práctica sobre los resultados y procedimientos aplicados.

- Las prácticas se basarán en ejercicios seleccionados y presentados de modo gradual en complejidad. La presentación de los ejercicios será guiada por los objetivos propuestos para el tema específico al cual la práctica se refiere. Los ejercicios serán seleccionados con un criterio que pondere lo conceptual y lo estratégico en lugar de la mecanización de procedimientos.

Trabajos Prácticos Por Unidad Temática

- Para poder realizar un seguimiento progresivo del aprendizaje, se asocian a las diversas unidades temáticas trabajos prácticos en la que los alumnos podrán aplicar lo aprendido.
- Estos trabajos prácticos posibilitan la resolución de problemas por unidad temática con objetivos propios, y consisten en planteos de problemas y actividades referentes a los diversos contenidos de la asignatura.
- El docente irá evaluando el progreso de cada alumno en cada entrega de los diferentes prácticos grupales o individuales.
- Se plantearán trabajos prácticos obligatorios y complementarios. Los docentes corregirán cada trabajo práctico entregado por los alumnos y darán una devolución personalizada.

Trabajos Prácticos Integradores

- Para poder realizar un aprendizaje integral de la aplicación de todos los contenidos de la materia se plantearán trabajos prácticos integradores obligatorios a los cuales se irán agregando poco a poco cada uno de los conceptos aprendidos durante la cursada.
- Esto trabajos estarán destinados a aplicar y medir el grado de comprensión de los temas teóricos expuestos en clase y el manejo de las definiciones y propiedades en contextos prácticos e integradores para comprobar que realmente se han incorporado los conceptos y no memorizado o mecanizado definiciones, procedimientos y demostraciones presentadas en las clases o que figuran en los libros.
- Los trabajos integradores tienen como finalidad generar la capacidad necesaria para saber interpretar claramente los objetivos del problema y poder resolverlo, aplicando una adecuada estrategia en la resolución.
- El alumno deberá ir realizando entregas parciales de avances establecidas por el docente durante la cursada. El docente hará seguimiento del alumno en cada entrega y exposición del práctico.

Materiales Didácticos

- La materia cuenta con apuntes teórico-prácticos desarrollados por los profesores de la cátedra. También se utilizan los libros detallados en la sección de Bibliografía.

Sitio Web: Sharepoint

- Sitio web destinado a facilitar al alumno el acceso al programa de la materia, material de estudio, ejemplos, trabajos prácticos, entre otros archivos y el contacto directo con docentes y alumnos.

EXPERIENCIAS DE LABORATORIO/ TALLER / TRABAJOS DE CAMPO:

Prácticas en Laboratorios: En cada una de las unidades se desarrollarán prácticas de laboratorios. En caso de no disposición de laboratorios y/o de forma complementaria, se darán ejercicios de referencia como guía para que el alumno pueda realizar su práctica en su propia PC.

Trabajo Práctico Integrador: Trabajo Práctico Integrador de todos los contenidos de la materia.

Índice de Prácticas:

Trabajo Práctico Parte 1	Presentación de Look & Feel con archivos HTML (aplicando estándar XHTML), hojas de estilos (CSS) y ASP.NET MVC
Trabajo Práctico Parte 2	Incorporar funcionalidad de Acceso a Base de Datos Entity Framework y Validaciones.
Trabajo Práctico Investigación	Investigar un tema brindado por la cátedra correspondiente a tecnologías .Net.

Software Utilizado:

- Microsoft Visual Studio 2012 o superior.
- Microsoft SQL Server 2008 o superior.

METODOLOGÍA DE EVALUACIÓN:

Exámenes Parciales

- Existirán dos evaluaciones parciales según lo indicado en el cronograma.
- Las evaluaciones serán con contenido teórico y-o práctico, pudiendo la cátedra llevar a cabo evaluaciones orales y/o en la PC.
- Los exámenes serán corregidos por los docentes del curso y las notas serán entregadas a los alumnos como máximo a los 7 días hábiles de la toma del parcial.

- Por cada examen parcial existirá un examen recuperatorio en fecha de recuperación.

Examen Final

- En el caso que el alumno cumpla con los requisitos establecidos en el Régimen de Cursada pero no con los criterios de promoción, deberá rendir un examen final.
- El primer llamado a examen final será al final del cuatrimestre según cronograma fijado por el Departamento de Ingeniería.
- Las fechas de examen final son fijadas por el Departamento de Ingeniería. Las condiciones de inscripción al final las establece el Departamento de Ingeniería.
- El examen final será confeccionado de forma uniforme para todas las comisiones.
- En fecha de final no se entregan trabajos prácticos.
- En el caso de exámenes libres se confeccionarán de forma especial de manera de evaluar la parte teórica/práctica con el mismo nivel que para alumnos regulares.
- Los exámenes serán corregidos por cualquier docente de la cátedra.

CRONOGRAMA ORIENTATIVO DE ACTIVIDADES

Clase	Contenido
1	Introducción General de la Materia (Asistencia, Parciales, TP) en Planilla Introducción a c#.
2	Introducción a .Net - IDE VS NetCore – NetStandard.
3	IDE Visual Studio, Consola, proyectos (Practica)
4	Introducción a ASP.NET. ASP.NET MVC Framework (VIEWS - Controllers- Models, Routing)
5	Introducción a ASP.NET MVC Framework (VIEWS - Controllers- Models, Routing) (Practica) Practica C# Practica de MVC, con Routing.
6	Razor, Vistas Tipadas y no tipadas (Teoría) MVC PRACTICA de Objetos: Mostrar un objeto Cliente con Razor. Mostrar una Lista de Objetos con Razor. Mapeo automático y manual sobre un Cliente. Lists - Generics
7	Practica MVC Razor sobre lista estática de objetos
8	POO (Teoría y Práctica) Practica con MVC
9	Ejercicios POO - C# Ejemplo Practica CRUD MVC
10	Principios POO Solid (SOLID) Teoría.

Clase	Contenido
11	Practica SOLID Validando Controles DataAnnotations (Teoría y Práctica)
12	Pasaje de datos (ViewBag - ViewData - Session - ViewModels) (Teoría) Layouts MVC, Partial Views,
13	Pasaje de datos (ViewBag - ViewData - Session - ViewModels) (Practica) ActionResults (redirecttoaction, redirect, file, json, etc.) Revisión de TP Investigación
14	Repaso Parcial
15	1er Parcial
16	Introducción ADO.NET ORM Entity Framework (EF)
17	Introducción EF - ORM, Arquitectura (Teoría y practica)
18	EF - Relaciones 1 a 1 y 1 a muchos (Teoría) LINQ (Teoría) LINQ To Entities (Teoría)
19	Entity Framework - Relaciones 1 a 1, 1 a muchos, muchos a muchos. (Teoría y práctica) LINQ Practica. LINQ To Entities Practica
20	EF - CRUD Completo Integrado Arquitectura 2 capas vs 3 capas (Teoría)
21	EF - CRUD Completo Integrado Arquitectura - 2 capas vs 3 capas (Practica)
22	Servicios Web / REST (Teoría)
23	Servicios Web / REST Practica
24	Entrega de TP's de Investigación
25	Revisión TP Entrega de TP's de Investigación
26	Repaso 2do Parcial
27	2do Parcial.
28	Entrega TP Completo – Defensa
29	Entrega TP Completo - Defensa
30	Recuperatorio 1er Parcial
31	Recuperatorio 2do Parcial
32	Cierre de notas

CONDICIONES DE CURSADA Y APROBACIÓN

Según lo establecido en la RHCS 054/2011 (Régimen académico integrado)

“Declaro que el presente programa de estudios de la asignatura Programación Web 3, es el vigente para el ciclo lectivo 2020, guarda consistencia con los contenidos mínimos del Plan de Estudios”

Firma

Aclaración

Fecha