

SILABO

I. INFORMACIÓN GENERAL

| | | |
|----------|---|------------------------------|
| Código | : | PDS-1902 |
| Pilar | : | Desarrollo de Software |
| Programa | : | Visual Studio 2017 Developer |
| Curso | : | Visual Studio 2017 Developer |
| Créditos | : | 6 |
| Duración | : | 96 horas |

II. DESCRIPCIÓN DEL CURSO

En este curso, el participante obtendrá los conocimientos y habilidades necesarias para el desarrollo de aplicaciones de escritorio con Windows Communication Foundation (WCF 4.6) y aplicaciones Web con ASP.NET WebForms y MVC. Además, aprenderá las características de las tecnologías ADO.NET, LINQ, Entity Framework 6.0 (EF) y Dapper para la gestión de la capa de datos que garanticen un buen performance de la aplicación.

III. PRE REQUISITOS

- Conocimiento de NET Framework 4.5.1 o 4.6 (nivel básico).
- Conocimiento de Visual Studio 2013 o superior (nivel básico).
- Conocimiento de Transact-SQL (nivel básico).
- Conocimiento de programación C#.NET (nivel intermedio).
- Conocimiento de programación orientada a objetos (nivel intermedio).
- Conocimiento de tipos de datos, variables, constantes y métodos (nivel intermedio).
- Conocimiento de interfaces al construir una clase (nivel intermedio).

CAPACITACIÓN RECOMENDADA

- Visual Studio 2015 Fundamentals
- Visual Studio 2017 Fundamentals

IV. LOGRO DEL CURSO

Al finalizar el curso, el alumno implementará aplicaciones de tipo escritorio y web utilizando el lenguaje de programación C#. Adicionalmente, estará preparado para:

- Utilizar correctamente las características de ADO.NET, LINQ, EF 6.0 y Dapper.
- Implementar servicios Windows y Web utilizando el patrón de diseño MVVM y MVC.
- Utilizar las herramientas de Reporting Services y componentes MS Office.

V. LÍNEA DE ESPECIALIZACIÓN



VI. CERTIFICACIONES

- Al término de la capacitación, el alumno obtendrá un certificado de Visual Studio 2017 Developer por aprobar el programa.
- Al completar satisfactoriamente 3 cursos de la línea de especialización, (ver punto V), se le acreditará un Diplomado en Desarrollo de Aplicaciones con Visual Studio.

VII. METODOLOGÍA

El enfoque práctico-reflexivo de cada una de las sesiones y la participación activa a lo largo de estas favorecerá no solo el aprendizaje de contenidos, sino también, su aplicación en la resolución de situaciones reales en diferentes contextos que se planteen. Por ello, el curso considera el desarrollo de laboratorios guiados e individuales, permitiendo la consolidación del aprendizaje del tema.

RECURSOS DE APRENDIZAJE

El recurso para el aprendizaje está compuesto por:

- Desktop/Laptop
- Silabo
- Marco teórico
- Diapositivas
- Laboratorios
- Lecturas
- Videos
- Tareas

VIII. EVALUACIÓN DEL CURSO

La calificación es vigesimal (0 a 20), siendo trece (13.00) la nota mínima aprobatoria. El promedio final del curso será el resultado de la siguiente fórmula:

$$PF = PP \times 0.20 + EP \times 0.30 + ET \times 0.20 + EF \times 0.30$$

LEYENDA

PF: Promedio final
PP: Promedio de prácticas (20 % de la nota final)
EP: Evaluaciones parciales (30 % de la nota final)
ET: Evaluaciones tareas (20 % de la nota final)
EF: Evaluación final (30 % de la nota final)

IX. CONTENIDO ANALÍTICO

| CAPÍTULO N° 1: Desarrollo de aplicaciones empresariales | | Duración: 6.0 h |
|---|--|----------------------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none">• Introducción al IDE Visual Studio• Componentes de .NET Framework• Tecnologías en la plataforma .NET• C#.NET y programación orientada a objetos• Nuevas características en C# | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none">• Conocer el ambiente de desarrollo de Visual Studio.• Identificar las características del .NET Framework.• Entender las tecnologías desarrolladas por Microsoft para la implementación de soluciones empresariales.• Comprender el uso de C#.NET y la POO. | |
| Laboratorios: | | |
| <ul style="list-style-type: none">• Laboratorio N°1: Laboratorio N°1: Desarrollar una aplicación básica con Visual Studio:<ul style="list-style-type: none">– Ejercicio 1.1: Crear una primera solución/proyecto utilizando C#.NET y POO. | | |
| Tarea: | | |
| <ul style="list-style-type: none">• Tarea N°1: Implementar una aplicación utilizando POO en Visual Studio. | | |

| CAPÍTULO N° 2: Implementación de acceso a datos (ADO.NET) | | Duración: 5.0 h |
|--|--|----------------------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none">• Definición de ADO.NET• Arquitectura de ADO.NET• NET Data Providers• Modelo de datos (conectado y desconectado)• Datasets y Datatables• Transacciones | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none">• Conocer la arquitectura de ADO.NET.• Identificar los NET Data Providers.• Utilizar las interfaces y clases del modelo de objetos de ADO.NET.• Implementar una conexión a la base de datos usando la clase SqlConnection.• Realizar consultas a la base de datos.• Entender las ventajas y desventajas de las estrategias de trabajo (conectado y desconectado). | |
| Laboratorios: | | |
| <ul style="list-style-type: none">• Laboratorio N°2: Desarrollar una capa de acceso de operaciones con ADO.NET:<ul style="list-style-type: none">– Ejercicio N° 2.1: Desarrollar una capa de acceso a los datos.– Ejercicio N° 2.2: Buscar un registro por la llave primaria la tabla.– Ejercicio N° 2.3: Buscar un registro.– Ejercicio N° 2.4: Seleccionar registros.– Ejercicio N° 2.5: Insertar un registro.– Ejercicio N° 2.6: Realizar transacciones locales. | | |
| Tareas: | | |
| <ul style="list-style-type: none">• Tarea N°2: Agregar funcionalidad a la capa de acceso a los datos con ADO.NET. | | |

| CAPÍTULO N° 3: Dapper Extensión | | Duración: 6.0 h |
|---|--|--------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> Definición de Dapper Uso de Dapper Mapeo con Dapper Transacciones con Dapper | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> Comprender el uso de un micro ORM (Dapper). Realizar operaciones de acceso de datos y manipulación de objetos con Dapper. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> Laboratorio N°3: Gestionar una transacción en la capa de acceso a datos: <ul style="list-style-type: none"> Ejercicio 3.1: Implementar Dapper en la capa de acceso a datos. Ejercicio 3.2: Realizar los métodos de búsqueda, inserción y eliminación con Dapper. Ejercicio 3.3: Implementar transacciones con Dapper. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> Tarea N°3: Agregar el soporte a transacciones en la capa de acceso a datos. | | |

| CAPÍTULO N° 4: Implementación de acceso a datos (Entity Framework 6.0) | | Duración: 12.0 h |
|---|--|---------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> Definición de Entity Framework (EF) Uso del enfoque Database First Uso del enfoque Code First Definición de LINQ Variantes de LINQ <ul style="list-style-type: none"> LINQ to Objects LINQ to Entities LINQ to XML | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> Comprender el uso de un ORM (EF). Realizar operaciones de acceso a datos y manipulación de objetos con Entity Framework. Entender el lenguaje integrado de consultas LINQ. Realizar operaciones de acceso a datos y manipulación de objetos con LINQ. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> Laboratorio N°4: Implementar la capa de acceso a datos utilizando EF y LINQ: <ul style="list-style-type: none"> Ejercicio 4.1: Instalar Entity Framework en proyectos DataAccess y DataAccess.Test. Ejercicio 4.2: Ejecutar consultas con Entity Framework. Ejercicio 4.3: Ejecutar sentencias con LINQ en un arreglo definido (LINQ to Objects) y en el modelo Entity Framework previamente generada (LINQ to Entities). | | |
| Tareas: | | |
| <ul style="list-style-type: none"> Tarea N°4: Agregar funcionalidad a la capa de acceso a los datos con Entity Framework. | | |

| CAPÍTULO N° 5: Entity Framework 6.0 avanzado | | Duración: 3.0 h |
|--|--|--------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> • Uso de un profiler con Entity Framework • Deferred execution vs Immediate execution • Lazy loading vs Eager Loading • Add / Attach • Revisión Entity Framework 7.0 | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Comprender el funcionamiento interno de Entity Framework. • Ejecutar consultas SQL con Entity Framework. • Aplicar configuración aplicada para manipular el contexto de datos. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> • Laboratorio N°5: Optimizar las operaciones de base de datos con Entity Framework: <ul style="list-style-type: none"> – Ejercicio 5.1: Monitorear consultas con Entity Framework. – Ejercicio 5.2: Implementar las sentencias de inserción y actualización. – Ejercicio 5.3: Ejecutar un Stored Procedure con Entity Framework. – Ejercicio 5.4: Ejecución diferida y ejecución inmediata. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> • Tarea N°5: Investigar características complementarias de Entity Framework (EF). | | |

| CAPÍTULO N° 6: Repositorio de datos | | Duración: 7.0 h |
|---|---|--------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> • Desarrollo en Capas • ¿Qué es el patrón repositorio? • ¿Qué es el patrón unit of work? • Implementando el patrón repositorio: Entity Framework | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Comprender la utilidad del patrón repositorio y unit of work. Implementar el patrón repositorio y unit of work. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> • Laboratorio N°6: Implementar el patrón repositorio: <ul style="list-style-type: none"> – Ejercicio 6.1: Refactorizar el proyecto de acceso a datos para poder aplicar el patrón repositorio. – Ejercicio 6.2: Implementar el patrón repositorio en el proyecto DataAccess. | | |
| Tarea: | | |
| <ul style="list-style-type: none"> • Tarea N°6: Incrementar repositorios a nuestro Unit of Work. | | |

| CAPÍTULO N° 7: Servicios Windows Communication Foundation (WCF) | | Duración: 12.0 h |
|---|--|---------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> • Evolución de objetos a servicios • ¿Qué es SOA? • ¿Qué es Windows Communication Foundation? (WCF) • Arquitectura de WCF • WCF Contracts <ul style="list-style-type: none"> – Service – Data – Message • Operadores Async/Await | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Implementar servicios WCF en una arquitectura orientada a servicios. • Definir e implementar contratos de datos y mensajes para servicios WCF. • Usar múltiples endpoints con patrones de mensajes variados. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> • Laboratorio N°7: Crear un servicio WCF: <ul style="list-style-type: none"> – Ejercicio 7.1: Agregar un nuevo método para la lectura de un procedimiento almacenado y mapear la información. – Ejercicio 7.2: Crear un servicio que expone un método para validar la factura de un cliente. | | |
| Tarea: | | |
| <ul style="list-style-type: none"> • Tarea N°7: Incrementar funcionalidad a nuestro servicio WCF, brindando más operaciones a consultar. | | |

| EVALUACIÓN PARCIAL 1 (EP) | Duración 2.0 hrs |
|---------------------------|---------------------|
|---------------------------|---------------------|

| CAPÍTULO N° 8: Arquitectura de IIS | | Duración 3.0 h |
|---|--|-------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> • Servidor IIS • Elementos del IIS • IIS Express | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Crear un sitio Web en la consola de administración del IIS. • Crear una aplicación Web haciendo uso de la consola de administración del IIS. • Conocer los principales elementos a configurar mediante la consola IIS. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> • Laboratorio N°8: Desplegar una aplicación ASP.NET en el IIS: <ul style="list-style-type: none"> – Ejercicio 8.1: Acondicionar la aplicación construida en laboratorios previos con IIS. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> • Tarea N°8: Determinar el funcionamiento de las opciones que se tienen en la configuración avanzada de los "Grupos de aplicaciones" (Application Pools) en IIS. | | |

| CAPÍTULO N° 9: Introducción a ASP.NET y WebForms | | Duración 7.0 h |
|--|---|-------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> • HTML5 • Introducción a ASP.NET • Introducción a WebForms • Controles ASP.NET • Controles de validación • Elementos de una aplicación ASP.NET <ul style="list-style-type: none"> • Carpeta Bin • Archivo Global.asax • Archivo de configuración Web.config | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Comprender las características de un WebForm. • Identificar los tipos de controles ASP.NET. • Crear un formulario usando controles Web ASP.NET. • Invocar un control de usuario. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> • Laboratorio N°9: Desarrollar una interfaz Web con ASP.NET WebForms: <ul style="list-style-type: none"> – Ejercicio 9.1: Crear un formulario Web Forms para visualizar la información de la entidad Products utilizando la capa de lógica de negocios. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> • Tarea N°9: Agregar una capa de presentación Web (WebForms) para realizar mantenimientos. | | |

| CAPÍTULO N° 10: Aplicando Master Pages, temas y estilos | | Duración 3.0 h |
|--|--|-------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> • Introducción y uso de MasterPages • Uso de estilos • Aplicando estilos a los controles | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Utilizar páginas maestras. • Utilizar temas en aplicaciones Web. • Aplicar estilos en páginas Web. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> • Laboratorio N°10: Gestionar la capa de presentación Web con MasterPages, temas y estilos: <ul style="list-style-type: none"> – Ejercicio 10.1: Crear un Master Page que brinde diferentes opciones a las que se tienen por defecto e incorporarla dentro de un nuevo WebForms. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> • Tarea N°10: Modificar la capa de presentación Web para incorporar un Master Pages, Temas y Estilos. | | |

| CAPÍTULO N° 11: Controles de datos en WebForms | | Duración 3.0 h |
|--|--|-------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> Introducción a los controles de visualización de datos <ul style="list-style-type: none"> DataList DetailsView FormView GridView Repeater DataSource y templates | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> Identificar los principales controles de visualización de datos, reconociendo sus características y funcionalidad. Utilizar apropiadamente las plantillas según el tipo de control. Establecer el origen de los datos de un control. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> Laboratorio N°11: Diseñar una aplicación Web con controles de datos: <ul style="list-style-type: none"> Ejercicio 11.1: Usar el control GridView y SqlDataSource para crear el mantenimiento de la tabla Artist. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> Tarea N°11: Agregar funcionalidad en base a los controles compuestos de ASP.Net WebForms: <ul style="list-style-type: none"> Tarea N°11.1: Modificar el control GridView para permitir darle mantenimiento a los elementos. Tarea N°11.2: Agregar nuevas páginas utilizando otros controles ASP.NET WebForms. | | |

| CAPÍTULO N° 12: Gestión de excepciones y seguimiento | | Duración 2.0 h |
|---|---|-------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> Tratamiento de errores estructurados y uso de Try/Catch Eventos Page_Error y Application_Error Configuración en el archivo Web.config Seguimiento de la aplicación Web | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> Implementar el manejo de error en las aplicaciones Web. Usar las trazas durante pruebas y depuración de la aplicación. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> Laboratorio N°12: Administrar y configurar errores en una aplicación Web. <ul style="list-style-type: none"> Ejercicio 12: Implementar excepciones personalizadas Try/Catch y configurar CustomErrors en el Web.Config. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> Tarea N°12: Consolidar conceptos sobre manejo de excepciones y notificación de estos en proyectos WebForms. | | |

| CAPÍTULO N° 13: Configurando seguridad de aplicaciones Web ASP.NET | | Duración 3.0 h |
|--|--|-------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> Autenticación y autorización ASP.NET Identity | Al finalizar el capítulo, el alumno podrá: <ul style="list-style-type: none"> Implementar tópicos de seguridad en las aplicaciones Web. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> Laboratorio N°13: Configurar la autenticación y autorización en aplicaciones Web ASP.NET: <ul style="list-style-type: none"> Ejercicio 13: Agregar autenticación y autorización a soluciones WebForms. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> Tarea N°13: Investigar sobre escenarios adicionales de autenticación y autorización en proyectos Web. | | |

| CAPÍTULO N° 14: Gestión y administración del estado de páginas Web | | Duración 3.0 h |
|---|---|-------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> Eventos de la aplicación <ul style="list-style-type: none"> A nivel de aplicación A nivel de sesión Modelo de objetos ASP.NET <ul style="list-style-type: none"> Application Session Caché Server Response Request | Al finalizar el capítulo, el alumno podrá: <ul style="list-style-type: none"> Identificar los principales elementos de una aplicación Web ASP.NET. Utilizar los objetos del modelo de objetos de ASP.NET. Identificar las diferentes formas de almacenamiento de sesiones. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> Laboratorio N°14: Gestionar el estado de una página Web: <ul style="list-style-type: none"> Ejercicio 14.1: Incorporar los objetos Session y Application en la aplicación ASP.NET | | |
| Tareas: | | |
| <ul style="list-style-type: none"> Tarea N°14: Consolidar los conocimientos sobre la gestión y administración del estado de páginas Web. | | |

| CAPÍTULO N° 15: Programación del lado del cliente | | Duración 7.0 h |
|---|--|-------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> ASP.NET AJAX Controles básicos de AJAX Instalación y uso de AJAX Toolkit Introducción a jQuery Usando jQuery en ASP.NET | Al finalizar el capítulo, el alumno podrá: <ul style="list-style-type: none"> Usar Ajax control tool kit. Utilizar JavaScript en aplicaciones Web. | |

| | |
|---|---|
| <ul style="list-style-type: none"> • Introducción a JSON • AJAX con jQuery | <ul style="list-style-type: none"> • Utilizar jQuery para simplificar el acceso al DOM. • Utilizar jQuery en páginas ASP.NET. |
| Laboratorios: | |
| <ul style="list-style-type: none"> • Laboratorio N°15: Diseñar una aplicación utilizando la programación del lado cliente: <ul style="list-style-type: none"> – Ejercicio 15.1: Agregar funcionalidad a la aplicación Web del lado del cliente con jQuery. | |
| Tareas: | |
| <ul style="list-style-type: none"> • Tarea N°15: Agregar funcionalidad del lado del cliente en una aplicación Web. | |

| | |
|----------------------------------|---------------------------------|
| EVALUACIÓN PARCIAL 2 (EP) | Duración 2.0 h |
|----------------------------------|---------------------------------|

| CAPÍTULO N° 16: Tecnologías complementarias en .NET | | Duración 6.0 h |
|--|---|---------------------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> • Diseño de reportes • Crystal Reports • Reporting Services • Desarrollo de soluciones de Office • Tipos de soluciones Office | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Publicar reportes. • Integrar reportes en formularios Windows. • Reconocer los tipos de aplicaciones que pueden integrarse a Office. • Crear un componente MS-Excel. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> • Laboratorio N°16: Implementar reportes con información estructurada sobre los datos que generan las aplicaciones: <ul style="list-style-type: none"> – Ejercicio 16.1: Crear reportes para visualizar información de invoices. | | |
| Tareas: | | |
| <ul style="list-style-type: none"> • Tarea N°16: Crear reportes utilizando los diferentes controles de Reporting Services. | | |

| CAPÍTULO N° 17: Pruebas unitarias | | Duración 6.0 h |
|--|---|---------------------------------|
| Temas | Habilidades | |
| <ul style="list-style-type: none"> • ¿Por qué las pruebas unitarias? • Desafíos con las pruebas unitarias • ¿Cómo se crea un unit test? • ¿Qué es un unit test? • Ventajas de los unit test | <p>Al finalizar el capítulo, el alumno podrá:</p> <ul style="list-style-type: none"> • Implementar pruebas unitarias al proyecto creado. | |
| Laboratorios: | | |
| <ul style="list-style-type: none"> • Laboratorio N°17: Optimizar los unit test creados con xUnit y FluentAssertions <ul style="list-style-type: none"> – Ejercicio 17: Instalar, configurar y usar xUnit con FluentAssertions para simplificar los unit test. | | |

| |
|---|
| Tareas: |
| <ul style="list-style-type: none"> • Tarea N°17: Investigar los patrones basados en Unit Test. |

| | |
|------------------------------|-----------------------------------|
| EVALUACIÓN FINAL (EF) | Duración 2.0 hrs |
|------------------------------|-----------------------------------|

X. REFERENCIAS BIBLIOGRÁFICAS

- JOYANES, Luis
2014 Fundamentos de Programación. Cuarta Edición. USA: MC GRAW HILL
- NEWSOME, Bryan
2012 *Beginning Visual Basic 2012*. Edición 2012. Indianapolis: John Wiley & Sons, Inc.
- STEPHENS, Rod
2012 *Visual Basic 2012 Programmer's Reference*. Edición 2012. Indianapolis: John Wiley & Sons, Inc.