

Arquitectura de Software

Código: SOA-101

Propuesta de Valor: DESARROLLO - PROGRAMACIÓN - METODOLOGÍAS

Duración: 30 Horas



En el ámbito del software cada vez es más común escuchar el término “arquitectura de software”, y encontrar oportunidades de empleo para “arquitectos de software”. Aún así, este concepto tiende a ser malentendido y la falta de comprensión al respecto de sus principios frecuentemente repercute de manera negativa en la construcción de sistemas de software.

El concepto de arquitectura de software se refiere a la estructuración del sistema que, idealmente, se crea en etapas tempranas del desarrollo. Esta estructuración representa un diseño de alto nivel del sistema que tiene dos propósitos primarios: satisfacer los atributos de calidad (desempeño, seguridad, modificabilidad), y servir como guía en el desarrollo.

AUDIENCIA

- Personas que deseen obtener conocimientos de Arquitectura de Software.

PRE REQUISITOS

- No tiene requisitos previos.

OBJETIVOS

- Obtenga las habilidades necesarias para tener éxito como diseñador y arquitecto de software.
- Comprender el papel de un arquitecto de software en el mundo digital.
- Comprender nuevos paradigmas de interacción, creatividad y métodos de diseño que muevan el enfoque del software a los humanos que lo usan.
- Explore las actividades de diseño y prueba de software que abordan los nuevos desafíos de los tiempos centrados en la tecnología.

CERTIFICACIÓN DISPONIBLE



CONTENIDO

1. MÓDULO : INTRODUCCIÓN ARQUITECTURA DE SOFTWARE

- 1.1. CONTEXTO
- 1.2. ESTEREOTIPOS
- 1.3. ANTECEDENTES HISTORICOS
- 1.4. DEFINICIÓN
- 1.5. CORRIENTES PRINCIPALES
- 1.6. CONCEPTOS FUNDAMENTALES
- 1.7. ESTILOS ARQUITECTONICOS
- 1.8. LENGUAJES DE DESCRIPCION ARQUITECTONICA (ADLS)
- 1.9. MÉTODOS CLÁSICOS, ARQUITECTÓNICOS Y ÁGILES
- 1.10. SITUACIÓN

2. MÓDULO : SCRUM (VISUAL STUDIO 2012 Y TEAM FOUNDATION SERVER 2012)

- 2.1. QUE ES SCRUM
- 2.2. REQUISITOS PARA PODER UTILIZAR SCRUM
- 2.3. CÓMO FUNCIONA SCRUM
- 2.4. FUNDAMENTOS DE SCRUM
- 2.5. LISTA DE OBJETIVOS
- 2.6. LISTA DE TAREAS DE ITERACIÓN
- 2.7. INTRODUCCIÓN A LA ESTIMACIÓN Y PLANIFICACIÓN ÁGIL
- 2.8. ESTIMACIÓN Y PLANIFICACIÓN DE ITERACIÓN BASADA EN COMPROMISO
- 2.9. MÉTRICAS Y CUADROS DE MANDO INTEGRAL PARA SCRUM

3. MÓDULO : FUNDAMENTOS DE ARQUITECTURA DE APLICACIONES

4. MÓDULO : EL PROCESO DE DISEÑO DE ARQUITECTURA

- 4.1. IDENTIFICAR LOS OBJETIVOS DE LA ARQUITECTURA
- 4.2. SELECCIONAR LOS CASOS DE USO ARQUITECTURALMENTE IMPORTANTES
- 4.3. REALIZAR UN ESQUEMA DEL SISTEMA
- 4.4. IDENTIFICAR LOS PRINCIPALES RIESGOS Y DEFINIR UNA SOLUCIÓN
- 4.5. CREAR ARQUITECTURAS CANDIDATAS
- 4.6. ASPECTOS DE DOMAIN DRIVEN DESIGN

5. MÓDULO : ARQUITECTURA MARCO N-CAPAS

- 5.1. ARQUITECTURA DE APLICACIONES EN N-CAPAS
- 5.2. ARQUITECTURA MARCO N-CAPAS CON ORIENTACIÓN AL DOMINIO

6. MÓDULO : CAPA DE INFRAESTRUCTURA DE PERSISTENCIA DE DATOS

- 6.1. CAPA DE INFRAESTRUCTURA DE PERSISTENCIA DE DATOS
- 6.2. ARQUITECTURA Y DISEÑO LÓGICO DE LA CAPA DE PERSISTENCIA DE DATOS
- 6.3. PRUEBAS EN LA CAPA DE INFRAESTRUCTURA DE PERSISTENCIA DE DATOS
- 6.4. CONSIDERACIONES GENERALES DE DISEÑO DEL ACCESO A DATOS
- 6.5. IMPLEMENTACIÓN EN .NET DE LA CAPA DE PERSISTENCIA DE DATOS

7. MÓDULO : CAPA DE MODELO DE DOMINIO

- 7.1. IMPLEMENTACIÓN DE LA CAPA DE DOMINIO CON .NET 4.0
- 7.2. EL DOMINIO
- 7.3. ARQUITECTURA Y DISEÑO LÓGICO DE LA CAPA DE DOMINIO

8. MÓDULO : CAPA DE APLICACIÓN

- 8.1. CAPA DE APLICACIÓN
- 8.2. ARQUITECTURA Y DISEÑO LÓGICO DE LA CAPA DE APLICACIÓN
- 8.3. COMPONENTES DE LA CAPA DE APLICACIÓN
- 8.4. IMPLEMENTACIÓN EN .NET DE CAPA DE APLICACIÓN

9. MÓDULO : CAPA DE SERVICIOS DISTRIBUIDOS

- 9.1. SITUACIÓN EN ARQUITECTURA N-CAPAS
- 9.2. ARQUITECTURAS ORIENTADAS A SERVICIOS Y ARQUITECTURAS EN N-CAPAS (N-LAYER)
- 9.3. SITUACIÓN DE ARQUITECTURA N-LAYER CON RESPECTO A APLICACIONES AISLADAS Y A SERVICIOS SOA
- 9.4. ¿QUÉ ES SOA?
- 9.5. PILARES DE SOA ('SERVICE ORIENTATION TENETS')
- 9.6. ARQUITECTURA INTERNA DE LOS SERVICIOS SOA
- 9.7. PASOS DE DISEÑO DE LA CAPA DE SERVICIOS
- 9.8. TIPOS DE OBJETOS DE DATOS A COMUNICAR
- 9.9. CONSUMO DE SERVICIOS DISTRIBUIDOS BASADO EN AGENTES
- 9.10. INTEROPERABILIDAD
- 9.11. RENDIMIENTO
- 9.12. COMUNICACIÓN ASINCRONA VS. SINCRONA
- 9.13. REST VS. SOAP
- 9.14. INTRODUCCIÓN A SOAP Y WS-*
- 9.15. ESPECIFICACIONES WS-*
- 9.16. INTRODUCCIÓN A REST
- 9.17. ODATA: OPEN DATA PROTOCOL
- 9.18. REGLAS GLOBALES DE DISEÑO PARA SISTEMAS Y SERVICIOS SOA
- 9.19. IMPLEMENTACIÓN DE LA CAPA DE SERVICIOS DISTRIBUIDOS CON WCF .NET 4.0
- 9.20. OPCIONES TECNOLÓGICAS
- 9.21. INTRODUCCIÓN A WCF (WINDOWS COMMUNICATION FOUNDATION)
- 9.22. IMPLEMENTACIÓN DE CAPA DE SERVICIOS WCF EN ARQUITECTURA N-LAYER
- 9.23. TIPOS DE OBJETOS DE DATOS A COMUNICAR CON SERVICIOS WCF
- 9.24. CÓDIGO DE SERVICIO WCF PUBLICANDO LÓGICA DE APLICACIÓN Y DOMINIO
- 9.25. DESPLIEGUE Y MONITORIZACIÓN DE SERVICIOS WCF EN WINDOWS SERVER APPFABRIC
- 9.26. REFERENCIAS GLOBALES DE WCF Y SERVICIOS

10. MÓDULO : CAPA DE PRESENTACIÓN

- 10.1. SITUACIÓN EN ARQUITECTURA N-CAPAS
- 10.2. NECESIDADES DE INVERTIR EN LA INTERFAZ DE USUARIO
- 10.3. NECESIDAD DE ARQUITECTURAS EN LA CAPA DE PRESENTACIÓN
- 10.4. PATRONES DE ARQUITECTURA EN LA CAPA DE PRESENTACIÓN
- 10.5. IMPLEMENTACIÓN DE CAPA DE PRESENTACIÓN
- 10.6. VALIDACIÓN DE DATOS EN LA INTERFAZ (WPF)
- 10.7. VALIDACIÓN DE DATOS EN LA INTERFAZ DE USUARIO (SILVERLIGHT)
- 10.8. IMPLEMENTACIÓN CON ASP.NET MVC 2.0

11. MÓDULO : CAPAS DE INFRAESTRUCTURA TRANSVERSAL

- 11.1. CAPAS DE INFRAESTRUCTURA TRANSVERSAL
- 11.2. SITUACIÓN DE INFRAESTRUCTURA TRANSVERSAL EN LA ARQUITECTURA
- 11.3. CONSIDERACIONES GENERALES DE DISEÑO
- 11.4. ASPECTOS TRANSVERSALES
- 11.5. IMPLEMENTACIÓN EN .NET DE ASPECTOS TRANSVERSALES

★ BENEFICIOS

- Al finalizar el curso, los participantes entenderán las distintas Fases que componen al Proceso de Arquitectura.