

DISEÑO DE SOFTWARE I



Identificación del curso.

Nombre del Ova o curso:	Diseño de software I
Programa:	Ingeniería del Software
Escritor:	Oralia Cortés G
Año, versión:	2014
Número de créditos:	3
Competencia Global del curso:	Realizar diseños de software que responden a las necesidades de un sistema de información considerando los atributos de calidad más relevantes para los usuarios

Estructura.

Elemento de competencia 1: Aplicar conceptos y técnicas de modelamiento de la vista conceptual de una arquitectura de software

TEMAS	HORAS	
	AC: 12	TI: 36
Qué es el análisis y diseño	2	6
Qué son los procesos de negocio	6	18
Lenguaje Unificado de Modelado	2	6
Diagrama de actividades	2	6

Nota: AC: Trabajo con acompañamiento docente. TI Trabajo independiente del estudiante.

Elemento de competencia 2: Aplicar conceptos y técnicas de modelamiento de la vista lógica de una arquitectura de software

TEMAS	HORAS	
	AC: 12	TI: 36

DISEÑO DE SOFTWARE I



Análisis de atributos de calidad	2	6
Análisis de requisitos	2	6
Diagramas de paquetes	2	6
Asignación de responsabilidades	2	6
Diagramas de clases	2	6
Asignación de responsabilidades	2	6

Nota: AC: Trabajo con acompañamiento docente. TI Trabajo independiente del estudiante.

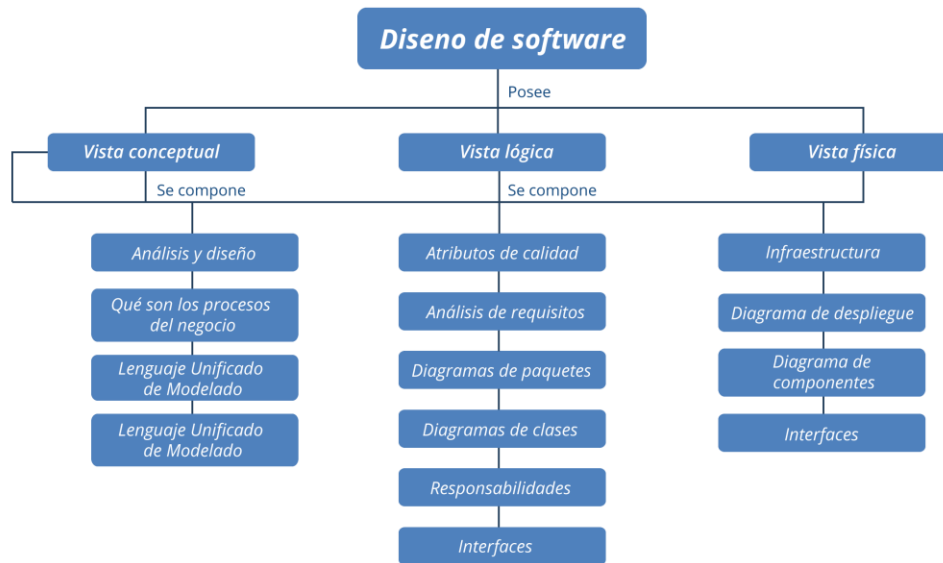
Elemento de competencia 3: Aplicar conceptos y técnicas de modelamiento de la vista física de una arquitectura de software

TEMAS	HORAS	
	AC: 12	TI: 36
Infraestructura	3	9
Diagrama de componentes	3	9
Diagrama de despliegue	3	9
Interfaces	3	9

Nota: AC: Trabajo con acompañamiento docente. TI Trabajo independiente del estudiante.

DISEÑO DE SOFTWARE I

Mapa Conceptual.



Metodología

El desarrollo del presente curso promueve el aprendizaje de tipo investigativo, recursivo, encaminado a la asimilación y ejecución práctica de los conocimientos y saberes adquiridos como posible solución a la situación problemática planteada.

Se plantea la continua necesidad de exploración de fuentes y recursos, actualización y continua investigación dentro de cada uno de los ejes temáticos propuestos, toda vez que el presente curso se desarrolla en modalidad virtual, aparte que se considera el Diseño de software como una actividad fundamental en el ciclo de vida del desarrollo de software.

Se hace vital y necesaria la continua interacción, comunicación y complementación entre el tutor y el estudiante, mediante los comentarios, aportes, preguntas aprovechando las herramientas virtuales de comunicación, a fin de recibir la oportuna orientación y complementación dentro del desarrollo de la situación o caso propuesto.

DISEÑO DE SOFTWARE I



El estudiante es el centro del proceso y su disciplina y autoestudio se constituyen en factores fundamentales para esta modalidad educativa. Adicionalmente las herramientas de la plataforma ofrecen espacios para la presentación de los contenidos, la interacción, la entrega de evidencias y verificación de competencias alcanzadas.

Las actividades de aprendizaje propuestas, permiten dejar plasmada como evidencia real de adquisición de conocimientos y habilidades, y gracias a la implementación de trabajos aplicativos, foros de discusión, video tutoriales y tutorías sincrónicas.

Evaluación

La evaluación por competencias en la Católica del Norte se orienta por los siguientes principios:

1. Continua: no hay momentos específicos para la evaluación, se realiza durante todo el periodo y a través de las actividades programadas en los módulos.
2. Integral: se valoran el ser, el saber y el hacer; es decir, el ser humano en todas sus dimensiones.
3. Refuerzo permanente: está implícito en el desarrollo de las actividades, porque el mismo criterio de desempeño se puede valorar en diferentes momentos. Los refuerzos culminan a la par de la terminación del bloque.

La evaluación de las competencias se realiza a partir de la entrega de las evidencias, éstas a su vez están planteadas como actividades y cada una cuenta con una serie de criterios que estandarizan la valoración de los resultados.

Los criterios de desempeño definidos para cada elemento de la competencia, son la base para determinar los resultados de aprendizaje que se estructuran con base en EVIDENCIAS DE APRENDIZAJE que son las pruebas manifiestas de aprendizaje, recogidas directamente durante el proceso formativo. Son recolectadas con la orientación del facilitador, utilizando técnicas, métodos e instrumentos de evaluación seleccionados, según sean evidencias de conocimiento, de producto o de desempeño, permitiendo reconocer los logros obtenidos

DISEÑO DE SOFTWARE I



por el estudiante en los tres tipos de saberes: conceptual, procedimental y actitudinal.

EVIDENCIAS DE CONOCIMIENTO. Apuntan al dominio cognoscitivo para procesar e identificar información relevante, su clasificación, su interpretación de manera útil, y la búsqueda de las relaciones entre información nueva e información adquirida previamente. Incluye el conocimiento de hechos y procesos, la comprensión de los principios, y teorías y las maneras de utilizar el conocimiento en situaciones cotidianas y nuevas.

EVIDENCIAS DE DESEMPEÑO. Evidencias del saber procedimental, relativas al cómo ejecuta el estudiante una actividad, en donde pone en juego sus habilidades, conocimientos y actitudes. Permiten recoger información directa, de mejor calidad y más confiable, sobre la forma como el estudiante desarrolla su proceso de aprendizaje y así poder identificar cuáles han sido sus logros y cuáles le faltan por alcanzar. Incluye las evidencias actitudinales.

EVIDENCIAS DE PRODUCTO. Son los resultados que obtiene el estudiante en una actividad que refleja el aprendizaje alcanzado y permite hacer inferencias sobre el proceso desarrollado, o método utilizado.

Justificación

La diseño de software es un área del conocimiento de la ingeniería de software relativamente joven, lo cual implica que sea difícil afrontar muchos tipos de problemas que no se han tratado antes, y lo que es peor, que muchos proyectos de software no culminen como se espera. De acuerdo con una encuesta del año 2004 realizada por Standish Group, solamente el 29% de los proyectos de software son exitosos y se realizan bajo los recursos estimados; la misma encuesta dice que el restante 71% son proyectos desviados o cancelados.

De igual manera, es importante tener en cuenta que el software hace cada vez más parte de nuestras vidas; y su relevancia juega cada vez un papel más crítico en la sociedad. Hoy en día no solamente se requiere que el software se entregue a los clientes dentro del tiempo pactado, sino que éste se caracterice por un conjunto de aspectos de calidad que garanticen la satisfacción de los clientes; y más significativo aún, que no ponga en riesgo el curso de un negocio o las personas que utilizan el software.

Es en el dominio de los aspectos de calidad donde la arquitectura y el diseño de software juegan el rol más

DISEÑO DE SOFTWARE I



importante. Desarrollar un software funcional no es una tarea compleja pero cuando éste debe incluir aspectos de alta disponibilidad, seguridad, rendimiento, confiabilidad, entre otros, es cuando se convierte en una tarea difícil y se puede convertir en imposible sino se aborda desde la arquitectura y el diseño de software.

Adicionalmente, la arquitectura y el diseño de software juegan un papel fundamental en la documentación de un sistema de información.

Por las razones previamente expuestas, se considera vital realizar una rigurosa y disciplinada fase de diseño de software dentro de un proyecto con el fin de garantizar que el sistema de información cumpla con sus tareas funcionales de la forma más adecuada; al igual que dejarlo documentado para su posterior evolución.

Durante el curso 'Diseño de Software I', se van a presentar los fundamentos y conceptos más relevantes dentro del área de la Diseño de Software; igualmente se van a presentar las técnicas utilizadas para modelar y documentar los productos de software, con el propósito que los estudiantes aprendan a aplicarlas dentro de un contexto real y adquieran habilidades de diseño de software de calidad.

Glosario

Ver archivo con glosario en la carpeta: [Glosario.docx](#)

Planteamiento del problema

Una empresa de software pequeña formada por estudiantes recién egresados de ingeniería informática está teniendo éxito con un producto de manejo de nómina y recursos humanos. Hasta el momento han tenido pocos pero buenos clientes, sin embargo, sus productos empiezan a llamar la atención de nuevos clientes; lo que ha hecho que se requiera personal adicional y software cada vez más personalizado para satisfacer las necesidades de los antiguos y nuevos clientes. Adicionalmente, la empresa está buscando el desarrollo de nuevos productos para su portafolio y de esta manera seguir creciendo.

Unos meses después de realizar algunos cambios estructurales referentes a la consecución de nuevo personal

DISEÑO DE SOFTWARE I



y expansión de mercado; la empresa está comenzando a tener diversos problemas, entre los más importantes:

- ✓ La curva de aprendizaje del nuevo personal es bastante alta, ya que no existe una documentación actualizada de los artefactos de software ni de la forma en cómo se construyen. Además, las personas con experiencia no cuentan con tiempo suficiente para entrenar al nuevo personal.
- ✓ La personalización del producto de software se dificulta en gran medida debido a que en el producto inicial no se consideraron tácticas de arquitectura apropiadas.

La empresa contrató los servicios de una firma de consultoría para ayudarle a atacar sus recientes problemas. La conclusión a la que se llegó es que la empresa debía levantar, adecuar y documentar la arquitectura de software de forma que cumpla con las expectativas de calidad de sus productos existentes y futuros.