



TECNOLÓGICO
NACIONAL DE MÉXICO



INSTITUTO TECNOLÓGICO[®]
de Pabellón de Arteaga
TEC

INGENIERIA DE SOFTWARE

25/05/2019

APRENDIENDO UML EN 24 HORAS



UNIFIED
MODELING
LANGUAGE

Docente:

Eduardo Flores Gallegos

Alumna:

Leslie Guadalupe Esparza Caldera

Carrera:

Ingeniería en Tecnologías de la Información y
Comunicaciones

Usuario de Windows

LESLIE ESPARZA



UML: Unified Modeling Language

El objetivo de UML es “proporcionar a desarrolladores de software, arquitectos de sistemas e ingenieros de software de herramientas para el análisis, diseño e implementación de sistemas basados en software, así como modelar procesos de negocio y similares. El modelado captura las partes esenciales del sistema. Introducción. UML es un lenguaje con un alcance muy grande y que cubre diversos conjuntos de dominios arquitectónicos en el diseño de aplicaciones. Por ello, no todas sus capacidades de modelados son necesariamente útiles en todos los dominios o aplicaciones. UML permite seleccionar sólo aquellas partes del lenguaje que sean realmente útiles.

- “El 80 por ciento de la mayoría de los problemas pueden modelarse usando alrededor del 20 por ciento de UML” - Grady Booch

Un objeto no es más que un conjunto de variables (o datos) y métodos (o funciones) relacionados entre sí. Los objetos en programación se usan para modelar objetos o entidades del mundo real.

herencia permite definir nuevas clases partiendo de otras ya existentes. Las clases que derivan de otras heredan automáticamente todo su comportamiento, pero además pueden introducir características particulares propias que las diferencian. La abstracción permite seleccionar las características relevantes dentro de un conjunto e identificar comportamientos comunes para definir nuevos tipos de entidades en el mundo real. La abstracción es clave en el proceso de análisis y diseño orientado a objetos, ya que mediante ella podemos llegar a armar un conjunto de clases que permitan modelar la realidad o el problema que se quiere atacar. El encapsulamiento permite reunir a todos los elementos que pueden considerarse pertenecientes a una misma entidad, al mismo nivel de abstracción. Polimorfismo Comportamientos diferentes, asociados a objetos distintos,



pueden compartir el mismo nombre, al llamarlos por ese nombre se utilizará el comportamiento correspondiente al objeto que se esté usando.

En base a su comportamiento, se pueden establecer varias categorías de objetos: Activos

– Pasivos Activo: posee un hilo de ejecución propio y puede iniciar una actividad

Pasivo: no puede iniciar una actividad, pero puede enviar estímulos una vez que se le solicita un servicio

Clientes – Servidores, Agentes Cliente es el objeto que solicita un servicio.

Servidor es el objeto que provee el servicio solicitado.

Una clase es una descripción de un conjunto de objetos que manifiestan los mismos atributos, operaciones, relaciones y la misma semántica. De igual manera también se podría decir que una clase es un conjunto de objetos que comparten una estructura y un comportamiento comunes. Contenedor de un tipo de datos asociados a un objeto (o a una clase de objetos), que hace los datos visibles desde fuera del objeto y esto se define como sus características predeterminadas, y cuyo valor puede ser alterado por la ejecución de algún método.

Agregación o acumulación. Los componentes y la clase que constituyen son una asociación que conforma un todo. Puede presentarse una agregación como una jerarquía dentro de la clase completa en la parte superior, y los componentes por debajo de la línea de ella. Una línea conectará el todo de un componente mediante un rombo sin relleno que se colocará en la línea más cercana del todo. Restricción de las agregaciones En ocasiones el conjunto de componentes posibles en una agregación se establece dentro de una relación O, para modelar esto, se utilizaría una restricción: la palabra O centro de llaves con una línea discontinua que conecte las dos líneas que conforman al todo. Composiciones Una composición es un tipo muy representativo de una agregación. Cada componente dentro de una composición puede pertenecer tan solo a un todo. El símbolo de una composición es el mismo que el de una agregación, excepto que el rombo está relleno. Contextos Cuando se modela un sistema se podría producir, con frecuencia,



TECNOLÓGICO
NACIONAL DE MÉXICO



agrupamiento de clases, como agregación y composiciones. En tal caso de, deberá enfocar toda la atención en un agrupamiento o en otro, y el diagrama de contexto le proporciona la característica de modelaje que requiere para tal fin. Las composiciones figuran en gran medida dentro de los diagramas de contexto. Un diagrama de contexto es como un mapa detallado de alguna sección de un mapa de mayores dimensiones. Pueden ser necesarias varias secciones para capturar toda la información detallada. Interfaces y realizaciones una interfaz es un conjunto de operaciones que especifica cierto aspecto de la funcionalidad de una clase, y es un conjunto de operaciones que una clase presenta a otras. (Captura el conjunto reutilizable.) La interfaz puede establecer un subconjunto de las operaciones de una clase y no necesariamente todas ellas para modelar una interfaz se hace con un símbolo rectangular, una interfaz no tiene atributos. Una forma es utilizar la estructura “estereotipo” y especificar la palabra sobre el nombre de la interfaz en el rectángulo. Otra es colocar la letra “I” al principio del nombre de una interfaz. La relación entre una clase y una interfaz se conoce como una realización. Esta relación esta modelada como una línea discontinua con una punta de flecha en forma de triangulo sin rellenar que adjunte y apunte a la interfaz. Otra forma (omitida) de representar una clase y su interfaz es con un pequeño círculo que se conecte mediante una línea a la clase. Una línea puede realizar más de una interfaz, y una interfaz puede ser realizada por más de una clase. Visibilidad este concepto esta muy relacionado con las interfaces y la realización. La visibilidad se aplica a atributos u operaciones de una clase dada (o en una operación de la interfaz). Existen tres niveles de visibilidad: nivel publico: en el cual la funcionalidad se extiende a otras clases. En el nivel protegido la funcionalidad se otorga solo a las clases que se heredan de la clase original. En el nivel privado solo la clase original puede utilizar el atributo u operación. La realización como podría imaginar, implica que el nivel público se aplique a cualquier operación en una interfaz. La protección de operaciones mediante cualquiera de los otros niveles tal vez no tendría sentido, dado que una interfaz se orienta a ser realizada por diversas clases. Para indicar el nivel publico, anteceda el atributo u operación con un signo de suma (+), para revelar un nivel protegido antecédalo con un símbolo (#), y para indicar el nivel privado antecédalo con un (-). Ambito Concepto referente a los atributos y operaciones, y la forma en que se relacionaba dentro de un sistema. Hay dos tipo de ámbitos, el de instancia y el de archivador. En el primero cada instancia cuenta con su propio valor en un atributo u

Usuario de Windows

LESLIE ESPARZA



TECNOLÓGICO
NACIONAL DE MÉXICO



operación. En un ámbito de archivador solo habrá un valor del atributo u operación en todas las instancias de las clases. Un atributo u operación con el ámbito de archivador, aparece con sus nombres subrayados. Este tipo de ámbito se utiliza con frecuencia cuando un grupo específico de instancias (ningunas otras) tiene que compartir los valores exactos de un atributo privado. Un caso de uso describe un conjunto de secuencias, en las cuales cada secuencia representa la interacción de cosas fuera del sistema (actores) con el sistema (y sus abstracciones clave). Estos comportamientos son funciones a nivel del sistema que acostumbra visualizar, especificar, construir y documentar en la fase de obtención y análisis de los requerimientos. Un caso de uso representa un (o más) requerimiento funcional completo del sistema. Por ejemplo, un caso de uso central de un banco es procesar préstamos. Los diagramas de caso de uso son centrales para modelar el comportamiento de un sistema, subsistema o una clase. Cada uno muestra un conjunto de casos de uso y actores y sus relaciones.

Aplicarás diagramas de caso de uso para modelar la vista de caso de uso de un sistema. Para la mayoría de las partes, esto involucra modelar el contexto de un sistema, subsistema o clase o modelar los requerimientos del comportamiento de estos elementos.

Los diagramas de caso de uso son importantes para visualizar, especificar y documentar el comportamiento de un elemento. Hacen al sistema, subsistema y clases accesibles y entendibles al presentar una vista externa de cómo aquellos elementos pueden ser usados en contexto. Sistemas de software intensivos pueden ser como lo anterior. Si eres un usuario, pudieras tener una aplicación y solicitarte que la uses. Si la aplicación sigue las convenciones normales del sistema operativo que acostumbras, puede ser que hagas algo útil después de un rato, pero no tendrás una comprensión de su comportamiento más detallado y complejo de esta manera. Similarmente, si eres un desarrollador, puede ser que tengas una aplicación heredada o un conjunto de componentes y te dicen que los uses. Estarás presionado para saber cómo usar los elementos hasta que te hayas formado un modelo conceptual de su uso. Los Diagramas de Estados representan autómatas de estados finitos. Son útiles sólo para los objetos con un comportamiento significativo el resto de objetos se puede considerar que tienen un único estado. Un diagrama de estado muestra

Usuario de Windows

LESLIE ESPARZA



TECNOLÓGICO
NACIONAL DE MÉXICO



la secuencia de estados que un objeto o una interacción pueden atravesar durante su existencia en respuesta a los estímulos que vayan recibiendo, junto con las correspondientes respuestas y acciones. Un diagrama de secuencias muestra la interacción de un conjunto de objetos de una aplicación a través del tiempo, en el cual se indicaran los módulos o clases que formaran parte del programa y las llamadas que se hacen cada uno de ellos para realizar una tarea determinada, por esta razón permite observar la perspectiva cronológica de las interacciones. Es importante recordar que el diagrama de secuencias se realiza a partir de la descripción de un caso de uso. El diagrama de colaboración es un tipo de diagrama de interacción cuyo objetivo es describir el comportamiento dinámico del sistema de información mostrando cómo interactúan los objetos entre sí. Los diagramas de actividad con un caso especial de las máquinas de estados. Si con éstas representábamos el comportamiento de un solo elemento, con los diagramas de actividad representamos procesos en los que interviene más de un clasificador y consideramos necesario representar cómo colaboran esos clasificadores entre sí.

Este libro tiene muy buenas formas de explicar con claridad todo acerca del UML, la forma en la que explica es tan bien, puesto que te deja en claro cada punto del cual habla. Me gusta el contenido y la forma en la que buscan que aprenda sea más fácil agregando algo didáctico al libro no refiriéndome a juegos o algo así si no que a ejemplos y preguntas que al ir leyendo surgen y al final de cada hora puedo responderlas gracias a las explicaciones antes ya dadas

Usuario de Windows

LESLIE ESPARZA