

Práctica #8

Disciplina de Diseño

1. Disciplina de Diseño

La disciplina de diseño busca desarrollar modelos enfocados en los requisitos no funcionales (tecnologías) y en el dominio de la solución, para prepararse para la implementación y pruebas del sistema. El objetivo es adquirir una comprensión profunda sobre los aspectos de los requisitos no funcionales y las restricciones.

La disciplina de diseño busca crear un diseño específico para una implementación. Crear un modelo físico, un esbozo de la implementación. Busca dar forma al sistema mientras intenta preservar la estructura definida por el modelo de análisis. Esta enfocado en los requisitos no funcionales, en hacer la aplicación correctamente, enfatizando una solución conceptual que cubra los requisitos más que en su implementación.

En RUP la disciplina de diseño genera como salida el Modelo de Diseño y el Modelo de Despliegue.

2. Modelo de Diseño

El modelo de diseño está muy cercano al de implementación. Es un modelo de objetos que describe la realización física de los casos de uso centrándose en cómo los requisitos funcionales y no funcionales, junto con otras restricciones relacionadas con el entorno de implementación, tienen impacto en el sistema a considerar. Además, el modelo de diseño sirve de abstracción de la implementación del sistema y es, de ese modo, utilizada como una entrada fundamental de las actividades de implementación.

Este modelo tiene los siguientes artefactos:

- Diagrama de clase de diseño
- Diagrama de paquetes de diseño
- Diagramas de interacción de los casos de uso

Una clase de diseño es una abstracción sin costuras de una clase o construcción similar en la implementación del sistema. Esta abstracción es sin costuras en el siguiente sentido:

- El lenguaje utilizado para especificar una clase del diseño es lo mismo que el lenguaje de programación. Se usa la sintaxis del lenguaje de programación escogido.
- Se especifica la visibilidad de los atributos y las operaciones
- Las relaciones de aquellas clases del diseño implicadas con otras clases, a menudo tienen un significado directo cuando la clase es implementada.

Un paquete de diseño es un subsistema. Los subsistemas de diseño son una forma de organizar los artefactos del modelo de diseño en piezas más manejables. Un subsistema puede constar de

clases del diseño, realizaciones de caso de uso, interfaces y otros subsistemas (recursivamente). Por otro lado, un subsistema puede proporcionar interfaces que representan la funcionalidad que exportan en término de operaciones. Las interfaces se utilizan para especificar las operaciones que proporcionan las clases y los subsistemas del diseño.

Por último, en el modelo de diseño, los casos de uso son realizados por las clases de diseño y sus objetos. Esto se representa por colaboraciones en el modelo de diseño y denota realización de caso de uso-diseño. Los objetos de diseño implicados en estas realizaciones interactúan para llevar a cabo el caso de uso. La manera de interactuar se refleja con diagramas de secuencia, ya que el centro de atención principal es el encontrar secuencias de interacciones detalladas y ordenadas en el tiempo.

Las realizaciones de caso de uso en los diferentes modelos sirven para cosas distintas. Cuando se diseñan las clases del análisis, especifican y hacen surgir clases de diseño más refinadas que se adaptan al entorno de implementación. Es útil crear un diagrama que muestre las clases de diseño del modelo de diseño con sus trazas hacia clases del modelo de análisis. La mayoría de las clases de diseño normalmente tienen una sola a una clase de análisis. Es habitual para las clases de diseño que son específicas de la aplicación tengan una sola traza a una clase de análisis. Pero también vemos surgir clases de diseño propias de las tecnologías escogidas que crean varias clases de diseño por una de análisis. Por lo tanto, la estructura del sistema definida por el modelo de análisis se conserva de forma natural durante el diseño, aunque pueden ser necesarios algunos cambios.

3. Modelo de Despliegue

El modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Este modelo se utiliza como entrada fundamental en las actividades de implementación dado que la distribución del sistema tiene una influencia principal en su diseño.

Podemos observar lo siguiente sobre el modelo de despliegue:

- Cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo hardware similar
- Los nodos poseen relaciones que representan medios de comunicación entre ellos, tales como Internet, intranet, bus, y similares
- El modelo de despliegue puede describir diferentes configuraciones de red, incluidas las configuraciones para pruebas y para simulación
- La funcionalidad (los procesos) de un nodo se definen por los componentes que se distribuyen sobre ese nodo
- El modelo de despliegue en sí mismo representa una correspondencia entre la arquitectura software y la arquitectura del sistema (el hardware)

Este modelo tiene un solo artefacto: el diagrama de despliegue.

4. Ejercicios

a) Cree el modelo de diseño del juego de mastermind, basado en el modelo de análisis hecho en la práctica 6 ejercicio a. Para esto realice lo siguiente:

1. Realizar el diseño de la arquitectura: identificar nodos y configuraciones de red y crear el diagrama de despliegue, identificar subsistemas y sus interfaces e identificar mecanismos genéricos de diseño (lenguaje programación, mecanismo de persistencia, seguridad, etc. lo que sea necesario)
2. Realizar el diseño de un caso de uso: identificar las clases del diseño de un caso de uso y describir las interacciones entre objetos de diseño, hacer la traza de las clases de diseño del modelo de diseño hacia clases del modelo de análisis.

b) Cree el modelo de diseño del sistema del taller mecánico, basado en el modelo de análisis hecho en la práctica 6 ejercicio b. Para esto realice lo siguiente:

1. Realizar el diseño de la arquitectura: identificar nodos y configuraciones de red y crear el diagrama de despliegue, identificar subsistemas y sus interfaces e identificar mecanismos genéricos de diseño (lenguaje programación, mecanismo de persistencia, seguridad, etc. lo que sea necesario)
2. Realizar el diseño de un caso de uso: identificar las clases del diseño de un caso de uso y describir las interacciones entre objetos de diseño, hacer la traza de las clases de diseño del modelo de diseño hacia clases del modelo de análisis.

c) Crear el diagrama de despliegue del modelo de diseño del proyecto. Crear también el diagrama de clases y secuencia de un caso de uso representativo de la arquitectura, así como el diagrama que muestre la traza de las clases de diseño del modelo de diseño hacia clases del modelo de análisis. Entregar los diagramas de despliegue, clases, secuencia y de traza del proyecto.

Al terminar, colocar en el reto del Aula Virtual el resultado del ejercicio “c” (el último), además coloque en el repositorio del proyecto en GitHub de su equipo, dentro del directorio “docs/logicalView/designView/” los archivos “.plantuml” del modelo de diseño y en “docs/physicalView” el modelo de despliegue.

Modifique el archivo “README.md”, coloque una sección llamada “Modelo de Diseño”, dentro de esa sección inserte las imágenes de los diagramas de clases y secuencia de los casos de uso, así como el diagrama de traza de las clases de diseño con las de análisis. Cree también una sección llamada “Modelo de Despliegue” e inserte el diagrama de despliegue del proyecto. La idea es tener un lugar donde se pueda consultar toda la información que se lleva del proyecto por modelo de RUP.

Suba todo a GitHub y coloque el URL del repositorio del proyecto en la entrega del reto 14. También puede subir a su repositorio personal de GitHub el resultado de los ejercicios a y b.