**DOCUMENTO DE TRABAJO TT7**

1. **IDENTIFICACIÓN**

**PROGRAMA ACADÉMICO**

|  |
| --- |
| **TALLER DE TÍTULO** |

|  |
| --- |
|  |

**CURSO**

|  |  |  |
| --- | --- | --- |
|  |  |  |

**CÓDIGO PLAN DE ESTUDIO**

|  |  |  |
| --- | --- | --- |
|  |  |  |

**REQUISITO(S) HORAS SEMANALES**

|  |
| --- |
|  |

**CARÁCTER**

**Elaborador: Magdalena Nieto G.**

**II. APRENDIZAJES ESPERADOS:**

El/la estudiante:

1. Elabora Modelo de Objetos para la capa Vista
2. Elabora Modelo de Objetos para la capa Controlador
3. Elabora la trazas de eventos

**III. SÍNTESIS ESQUEMÁTICA DE CONTENIDOS**

Diseño de

objetos

Modelo de

Objetos de la

capa Vista

Modelo de

objetos de la

capa Controlador

-

Diseño de

comportamiento

Traza de eventos

# **Diseño orientado a objeto**

**OBJETIVO DE LA FASE DEL DISEÑO**

El diseño tiene por finalidad especificar cómo será implementada la solución. Define los **métodos** de las clases

**ACTIVIDADES DE LA FASE DE DISEÑO**

* Diseño de objeto
* Diseño de comportamiento

1. **DISEÑO DE OBJETOS**

Las clases de la **vista** se obtienen fácilmente de la fase de especificación de la interfaz usuaria. Cada diálogo del diagrama de diálogos se transforma en una clase vista. Cada componente especificada en los diagramas de diálogo es una componente vista.

Las clases de la capa **controlador** conectan la vista con el modelo de objetos. Por cada objeto vista existe un único objeto controlador. Un objeto controlador puede estar ligado a múltiples objetos del modelo y un objeto del modelo puede estar ligado a varios objetos controladores.

1.1. Modelo de objetos de la capa VISTA (Interfaz, pantallas)

1.2. Modelo de Objetos de diseño de la capa CONTROLADOR.(Código, RAM)

* 1. **Modelo de objetos de la capa VISTA**

Cada diálogo de la interfaz gráfica representa un objeto compuesto de atributos y métodos (observe cuidadosamente cada diálogo y en él encontrará atributos (datos) y métodos (funciones) que los dispone dentro de la clase.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sistema de Biblioteca** |  | **Información de Alumno** |  | **Préstamo de Material** |
| Fecha  Hora |  | Num-rut  Nombres  Ap-paterno  Ap-materno  Carrera-Alumno  Inf-Atrasos |  | Cod-material  Título  Autor  Estado |
| ObtFecha()  SelInformacionAlum()  SelDevolucion()  SelInformes()  SelMantencionMat()  Salir() |  | BuscarInfAlum()  IrPrestamo()  IrConsulta()  IrReserva()  Salir() |  | Buscar()  AsignarPrestamo()  Salir() |
|  |  |  |  |  |
| **Consulta en sala** |  | **Reserva** |  | **Devolución de Material** |
| Cod-material  Título  Autor  Estado |  | Cod-material  Título  Autor  Estado |  | Cod-material  Estado-material |
| Buscar()  AsignarConsulta()  Salir() |  | Buscar()  AsignarReserva()  Salir() |  | Buscar()  Devolucion()  Salir() |
|  |  |  |  |  |
| **Mantenedor de Material** |  | **Informes de Material** |  | **Mensaje** |
| Cod-material  Título  Autor  Tipo  Detalle |  | Fecha  Periodo  Texto |  | Mensaje |
| Buscar()  Nuevo()  Eliminar()  Modificar()  Salir() |  | Ver()  Imprimir()  Salir() |  | SelAceptar() |

Ilustración 1: Modelo de objetos del diseño (capa VISTA).

## 1.2. Modelo de Objetos de diseño de la capa CONTROLADOR.

En la fase de Análisis OO, se elaboró un primer Modelo de Objetos, pero en ese entonces lo sustantivo fue identificar cuáles eran los objetos para la aplicación, pues bien, ahora en la fase de Diseño OO, tomamos ese Modelo elaborado en el AOO y agregamos los atributos y métodos que ellos deben contener, bajo la perspectiva de los objetos que manipulará el CONTROLADOR. esto siempre y cuando trabaje con Base de Datos OO, de los contrario, si trabajas con Modelo de Base de Datos Relacional DEBE reemplazar este Modelo de Objetos por el Modelo de Datos de su Base de Datos Relacional

**Bibliotecaria**

Nombre

Clave

Validar

Emite

Mantiene

**Informe**

Desde

Hasta

Ver

Imprimir

1 1..\*

CU1:Logear

CS7:Emite

1

1

CU8:Mantiene

1

CU2:ValidaAlum

1..\* 1..\*

**Material**

Codigo

Titulo

Autor

Asigna

**Alumno**

Rut

Nombre

Carrera

Estado

Solicita

Consulta

Reserva

Devuelve

CU3:Prestamo

CU4:Consulta

CU5:Reserva

CU6:Devuelve

1..\* 1..\*

Ilustración 2: Primer modelo de objetos del diseño. (capa CONTROLADOR, conecta con la BD)

1. **Diseño del comportamiento**

**2.1. Elaboración de Trazas de Eventos (diagramas de secuencia).**

El diseño orientado al objeto se compone de: las vistas el controlador y el modelo.

Las vistas es lo que el usuario ve y con lo que interactúa. Al ejecutar alguna acción el usuario provoca que la vista la interprete y le pida requerimientos al controlador.

Por cada operación definida en la etapa de análisis se debe generar su traza de eventos o diagrama de secuencia.

Al dibujar las trazas de eventos se especifican las responsabilidades de los objetos del diseño, en definitiva las funciones que llevarán a la práctica.

El nombre de la función es escrito sobre la flecha. Las llamadas a funciones se representan a través de flechas sobre la cual se escribe el nombre de la función y cuando corresponde se señalan los parámetros necesarios entre paréntesis, y los valores retornados de las llamadas a funciones son escritos sin paréntesis.

Las trazas de eventos grafican las hebras de ejecución como normalmente se han de realizar.

A continuación se inicia la construcción de las trazas de eventos para las operaciones especificadas en los casos de usos representando la responsabilidad del Actor y del Sistema

Cada caso de uso debe elaborar una traza de eventos.

También por cada flujo alternativo debe elaborar su traza de eventos

|  |  |  |
| --- | --- | --- |
| **Caso de uso: Validar información del alumno** | |  |
| **Flujo normal o escenario exitoso** | | |
| **Responsabilidad del Actor** | **Responsabilidad del Sistema** | |
|  | 1. **Despliega ventana de Información del alumno** (Habilita el ingreso del rut del alumno y el botón “buscar” y el botón “salir”) | |
| 1. **Ingresa rut** 2. **Acepta rut** (botón Buscar) |  | |
| 1. **Selecciona un caso de uso** (“Préstamo”, “Consulta” o “Reserva”) | 1. **Valida rut** (valida el ingreso del rut) 2. **Valida alumno** (valida que sea usuario de biblioteca, lee la BD) 3. **Despliega información-alumno** (presenta datos del alumno y activa los casos de uso “Préstamo”, “Consulta” y “Reserva” )      1. **Fin del caso de uso** (El actor es derivado al caso de uso seleccionado) | |

**Teclado Pantalla/interfaz RAM/Código Base de Datos**

**Modelo**

**Usuario**

**Controlador**

**Vista**

**1** **VtnInformAlum( )**

* Los flujos entre las 3 capas NO debieran estar cortados, deben permitir un tránsito.
* Si la última operación del Sistema es derivar a otro caso de uso, no es necesario que aparezca como flujo, va dentro de lo que se entiende como fin del caso de uso

**2** **Ingresa (rut)**

En este sentido la flecha

Cuando se usa teclado

**3 PresionaAcepta (rut)**

* Los nombres de métodos, no tienen por qué ser tan extensos, recuerde que un método es sinónimo de una function.
* Siempre el método debe llevar ( ). Si llevan datos deben ir entre los paréntesis, pues representan parámetros.
* Toda vez que se lee en la BD retorna datos leídos
* Siempre que se actualiza (agrega, modifica o elimina) la BD, esta retorna un flag que señala operación exitosa

Este sentido la flecha al usar mouse

**4** **ValidaRut(rut)**

Este sentido la flecha al Trabaja sobre RAM

**5** **ValidaAlumnoEnBD(rut)**

Este sentido la flecha al ir por datos a la Base de Datos

**5 DevuelveDatosLeidos(alumno)**

BD retorna los datos leídos

Si se ha actualizado la BD, retorna un flag de éxito

**6 6 DespliegaDatos(alum)**

**7 SeleccionaCaso( )**

* 1. **FinCasoUso( )**

Ilustración 3: Traza de eventos

|  |  |
| --- | --- |
| **Flujo alternativo 1: Valida el ingreso del rut (digitación). Si rut fue mal ingresado** | |
| **Responsabilidad del Actor** | **Responsabilidad del Sistema** |
| 1. **Acepta el mensaje** (botón Aceptar) | 1. **Despliega mensaje** (“rut mal ingresado” y habilita botón Aceptar) 2. **Fin flujo alternativo 1** |

**Vista**

**Usuario**

**Modelo**

**Controlador**

**1** **Mensaje(“rut mal ingresado”)**

**2** **AceptaMensaje ( )**

**3 FinFlujoAlternativo( )**

Ilustración 4: Traza de eventos

|  |  |
| --- | --- |
| **Flujo alternativo 2: Valida que alumno sea usuario de biblioteca. Si no es usuario** | |
| **Responsabilidad del Actor** | **Responsabilidad del Sistema** |
| 1. **Acepta el mensaje** (botón Aceptar) | * 1. **Despliega mensaje** (“alumno no registrado como usuario de biblioteca” y habilita botón Aceptar)  1. **Fin flujo alternativo 2** |

**Controlador**

**Vista**

**Modelo**

**Usuario**

**1** **Mensaje(“No registrado”)**

**2** **AceptaMensaje ( )**

**3 FinFlujoAlternativo( )**

Ilustración 5: Traza de eventos

**IV. ACTIVIDADES (INDIVIDUALES O GRUPALES)**

**Actividades 1: Elabore el Diseño de objetos**

* + - Modelo de objetos de la capa VISTA
    - Modelo de objetos de la capa CONTROLADOR

**Actividades 2: Elabore el Diseño de comportamiento**

* + - Traza de eventos (Diagrama de secuencias)

**V. EVALUACIÓN DE LA ACTIVIDADES**

**La actividad será evaluada de acuerdo con la siguiente escala de apreciación**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Descriptor** | **1** | **2** | **3** | **4** |
| Elabora el Modelo de objetos de la capa VISTA  Representados todos los diálogos donde cada uno de ellos representa un objeto compuesto de atributos y métodos |  |  |  |  |
| Elabora el Modelo de objetos de la capa CONTROLADOR  Representadas todas las tablas que componen a la Base de Datos Relacional, con sus relaciones |  |  |  |  |
| Elabora las Trazas de eventos (Diagrama de secuencias)  Representados todas las operaciones del caso de uso que se diagrama, tanto las de responsabilidad del actor como las de responsabilidad del sistema |  |  |  |  |
| **1: No cumple con el descriptor**  **2: Cumple con el descriptor de forma deficiente con omisiones o errores**  **3: Cumple con el descriptor con errores menores**  **4: Cumple cabalmente con el descriptor** | | | | |

**VI. SÍNTESIS DE LOS APRENDIZAJES:**

Los conceptos más importantes que aprendí esta semana son:

|  |
| --- |
|  |

Lo que puedo aplicar en mi trabajo, o en mi rutina personal o familiar es:

|  |
| --- |
|  |

Las actitudes que desarrollé esta semana son:

|  |
| --- |
|  |

**VII. GLOSARIO:**

**Capa vista:**

Representa a la interfaz gráfica de la aplicación, que se visualiza por pantalla

**Capa controlador:**

Representa al código de la aplicación , que se ejecuta en RAM

**Capa modelo**

Representa a los datos de la Base de Datos

|  |
| --- |
|  |

**VIII. LINKS DE INTERÉS:**

[**http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r11\_art4\_c.pdf**](http://www.unab.edu.co/editorialunab/revistas/rcc/pdfs/r11_art4_c.pdf)

[**http://sistemas.itlp.edu.mx/tutoriales/fundamentosdeprog/t21.htm**](http://sistemas.itlp.edu.mx/tutoriales/fundamentosdeprog/t21.htm)