

Modelo de Diseño

<Nombre del Proyecto>

Nombre del Grupo de Desarrollo o Asignatura

Nombre del Autor

**

**

*[Este documento es la plantilla base para elaborar el documento Modelo de Diseño.*

*Los textos que aparecen entre paréntesis rectos son explicaciones de que debe contener cada sección, los cuales se encuentran con estilo “PSI – Comentario”. Dichos textos se deben seleccionar y sustituir por el contenido que corresponda en estilo “Normal”.*

*Para actualizar la tabla de Contenido, haga clic con el botón derecho del ratón sobre cualquier línea del contenido de la misma y seleccione Actualizar campos, en el cuadro que aparece seleccione Actualizar toda la tabla y haga clic en el botón Aceptar.]*



**Tabla de contenido**

[**Introducción 4**](#_gjdgxs)

[*Propósito 4*](#_30j0zll)

[*Alcance 4*](#_1fob9te)

[*Definiciones, siglas y abreviaturas. 4*](#_3znysh7)

[**Referencias 4**](#_2et92p0)

[*Visión general 4*](#_tyjcwt)

[**Diseño de Casos de Uso 5**](#_3dy6vkm)

[*Diseño del Caso de Uso [nombre del caso de uso 1] 5*](#_1t3h5sf)

[Diagrama de paquetes 5](#_4d34og8)

[Diagrama de Interacción 5](#_2s8eyo1)

[Diseño de Flujo de eventos 5](#_17dp8vu)

[Requerimientos especiales o de implementación 5](#_3rdcrjn)

[*Diseño del Caso de Uso [nombre del caso de uso 2] 6*](#_26in1rg)

[Diagrama de paquetes 6](#_lnxbz9)

[Diagrama de Interacción 6](#_35nkun2)

[Diseño de Flujo de eventos 6](#_1ksv4uv)

[Requerimientos especiales o de implementación 6](#_44sinio)

[*Diseño de Objetos 6*](#_2jxsxqh)

[*[Objeto 1] 6*](#_z337ya)

[*[Objeto 2] 6*](#_3j2qqm3)

[**Diseño de Subsistemas 7**](#_1y810tw)

[*Subsistemas Específicos 7*](#_4i7ojhp)

[[Nombre del Subsistema Específico 1] 7](#_2xcytpi)

[Propósito 7](#_1ci93xb)

[Función 7](#_3whwml4)

[Subordinados 7](#_2bn6wsx)

[Dependencias 7](#_qsh70q)

[Recursos 8](#_3as4poj)

[**Diagramas 9**](#_1pxezwc)

[*Diagrama de componentes 9*](#_49x2ik5)

[*Diagrama de Clases 9*](#_2p2csry)

[*Diagrama de Secuencia 10*](#_147n2zr)

[*Diagramas de Paquetes 11*](#_3o7alnk)

[*Diagrama de Colaboración 12*](#_23ckvvd)

Modelo de Diseño

**Introducción**

*El modelo de Diseño es un refinamiento y formalizacion, donde se toman en cuenta las consecuencias del ambiente de implementacion. El resultado son especificaciones muy detalladas de todos los objetos, incluyendo sus operaciones y atributos. El modelo de Diseño se basa en el diseño por responsabilidades.*

**Propósito**

*[Esta sección debe indicar el propósito del documento Modelo de Diseño y la audiencia esperada para este documento.]*

*Se requiere un modelo de diseño, ya que el modelo de analisis no es lo suficientemente formal para alcanzar el codigo fuente. Por tal motivo se refinan los objetos, incluyendo las operaciones y atributos. También otro objetivo, es validar los resultados de los modelos de requisitos y análisis. Durante el diseño, se ve si los resultados anteriores son apropiados para la implementación. Si se descubren aspectos que no están claros en alguno de los modelos anteriores, estos se aclaran, posiblemente regresando a etapas anteriores.*

*Como el modelo de analisis define la arquitectura general del sistema, se busca obtener la arquitectura detallada como resultado del modelo de diseño, de manera que haya una continuidad de refinamiento entre los 2 modelos*

*las estructuras con las cuales se trabaja en el modelo de diseño son basicamente las mismas que en el modelo de analisis. Sin embargo , el punto de vista cambia, ya que se toma un paso hacia la implementacion. El modelo de analisis debe verse como un modelo conceptual y logico del sistema, en tanto el modelo de diseño debe acercarse al codigo fuente. Esto significa que se cambia el punto de abstraccion del código fuente final. Por lo tanto, el modelo de diseño debe ser una descripción de como debe estructurarse, administrarse y escribirse el código fuente.*

**Alcance**

*[Esta sección es una breve descripción de a que se aplica el documento Modelo de Diseño, que es afectado o influenciado por este documento.]*

**Definiciones, siglas y abreviaturas.**

*[Esta sección debe proporcionar las definiciones de todos los términos, las siglas, y abreviaturas requeridas para interpretar apropiadamente el documento Modelo de Diseño. Esta información puede proporcionarse por la referencia al Glosario del proyecto.]*

**Referencias**

*[Esta sección debe proporcionar una lista completa de todos los documentos a los que se hace referencia en el documento Modelo de Diseño. Cada documento debe identificarse por el título, número del informe (si se aplica), fecha, y organización que lo publica. Especifique las fuentes de las que pueden obtenerse las referencias. Esta información puede proporcionarse por la referencia a un apéndice o a otro documento.]*

*Modelo de Especificacion de Caso de Usos*

*Modelo de Datos*

**Visión general**

*[Esta sección describe que contiene el resto del documento Modelo de Diseño y explica como se organiza este documento.]*

**Diseño de Casos de Uso**

*[En esta sección se especifica el comportamiento de los casos de uso, mediante subsistemas u objetos de diseño que interactúan, y se determinan las operaciones e interfaces de los distintos subsistemas u objetos de diseño.*

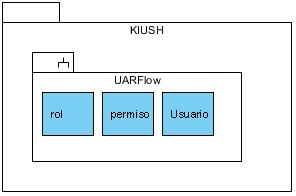
*Un objeto de diseño es una abstracción de un objeto X o varios objetos X en la implementación del sistema. Se define en base a parámetros, reglas, operaciones que realiza, requerimientos de implementación, interfaz de usuario (si corresponde) y relaciones con otros objetos.*

*Esta especificación se realiza mediante Diagrama de Paquetes, Diseño de flujo de eventos, Diagramas de interacción y requerimientos especiales o de implementación.]*

**Diseño del Caso de Uso [nombre del caso de uso 1]**

**Diagrama de paquetes**

*[En esta sección se identifican los objetos y subsistemas de diseño que intervienen en el caso de uso y sus relaciones.]*

**

**Diagrama de Interacción**

*[Se sugiere realizar un diagrama de interacción para representar la realización de cada Caso de Uso.]*

**Diseño de Flujo de eventos**

*[Descripción escrita en términos de objetos y subsistemas de diseño, que explica y complementa el diagrama de interacción y sus niveles.]*

**Requerimientos especiales o de implementación**

*[Descripción que recoge los requerimientos (no funcionales) en la realización de un caso de uso.]*

**Diseño del Caso de Uso [nombre del caso de uso 2]**

**Diagrama de paquetes**

*[En esta sección se identifican los objetos y subsistemas de diseño que intervienen en el caso de uso y sus relaciones.]*

**Diagrama de Interacción**

*[Se sugiere realizar un diagrama de interacción para representar la realización de cada Caso de Uso.]*

**Diseño de Flujo de eventos**

*[Descripción escrita en términos de objetos y subsistemas de diseño, que explica y complementa el diagrama de interacción y sus niveles.]*

**Requerimientos especiales o de implementación**

*[Descripción que recoge los requerimientos (no funcionales) en la realización de un caso de uso.]*

**Diseño de Objetos**

**[Objeto 1]**

*[Se especifican los parámetros, reglas, condiciones usando la misma sintaxis del lenguaje y el código y métodos son especificados en pseudocódigo.]*

* *Descripción:*
* *Objetivo:*
* *Atributos:*
* *Métodos*
* *Interacciones:*
* *Diagrama de estado*
* *Interfaces que implementa*

**[Objeto 2]**

*...*

**Diseño de Subsistemas**

**Subsistemas Específicos**

**[Nombre del Subsistema Específico 1]**

**Propósito**

*[Descripción de por qué el subsistema existe. Este atributo debe dar la razón de la creación del subsistema. Como ser la funcionalidad específica y los requerimientos de performance por los cuales fue creado. También describe requerimientos especiales que se deben lograr con él que no están incluidos en la especificación de requerimientos del software.]*

**Función**

*[Expresa qué realiza el subsistema. Establece la transformación aplicada a las entradas del subsistema para producir la salida deseada.]*

**Subordinados**

*[Se identifican los objetos de diseño y subsistemas de diseño que componen el subsistema que se describe. Se propone representar esta información con un diagrama de paquetes.]*

**Dependencias**

*[Descripción de la relación de este subsistema con otros subsistemas de diseño. Describir la naturaleza de cada interacción incluyendo características como tiempo y condiciones de la interacción. Estas, pueden involucrar la iniciación, orden de ejecución, datos compartidos, creación, duplicación, uso o almacenamiento.*

*Se propone representar esta información con una tabla de dependencias.]*

|  |  |  |
| --- | --- | --- |
| Subsistema del que depende | Naturaleza de interacción | Características |
| *[Identificación del subsistema del que depende]* | *[Condiciones para que se realice la interacción]* | *[Características de la interacción, como ser, pasaje de parámetros, mensajes, datos compartidos, etc.]* |

**Recursos**

*[Se deben Identificar y describir todos los recursos externos al diseño que necesita el subsistema para realizar su función. Especificar las reglas de interacción y métodos para usar el recurso.*

*Este atributo brinda información sobre elementos como dispositivos físicos (impresoras, particiones de disco, bancos de memoria), servicios de software (librerías, servicios del sistema operativo), y recursos de procesamiento (ciclos de CPU, ubicación de memoria, buffers).*

*Se deben describir características de uso como el tiempo de proceso al cual se debe adquirir el recurso e incluir la cantidad de tiempo de uso. Debe incluir también la identificación de capacidad potencial y facilidades de manejo del recurso.]*

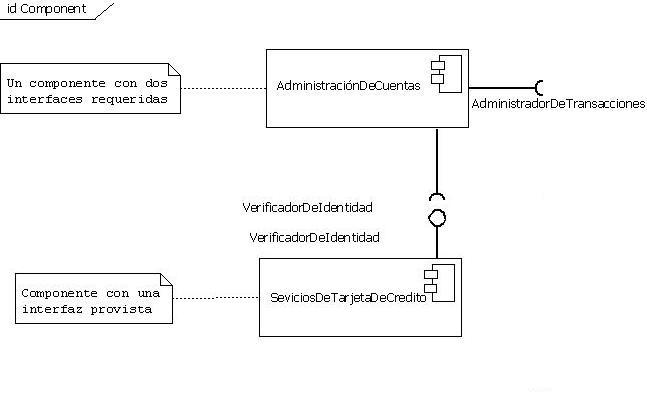
# Diagramas

**Diagrama de componentes**

*[Lo que distingue el diagrama de componentes de otro tipo de diagramas es sin duda su contenido. Normalmente contiene componentes, interfaces y relaciones entre ellos.*

*Los componentes pertenecen a un mundo físico, es decir, representan a un bloque de construcción al modelar aspectos físicos de un sistema.*

*Cada componente debe tener un nombre que lo distinga de los demás. Al igual que las clases los componentes pueden enriquecerse con compartimientos adicionales que muestran sus detalles. ]*

**

**Diagrama de Clases**

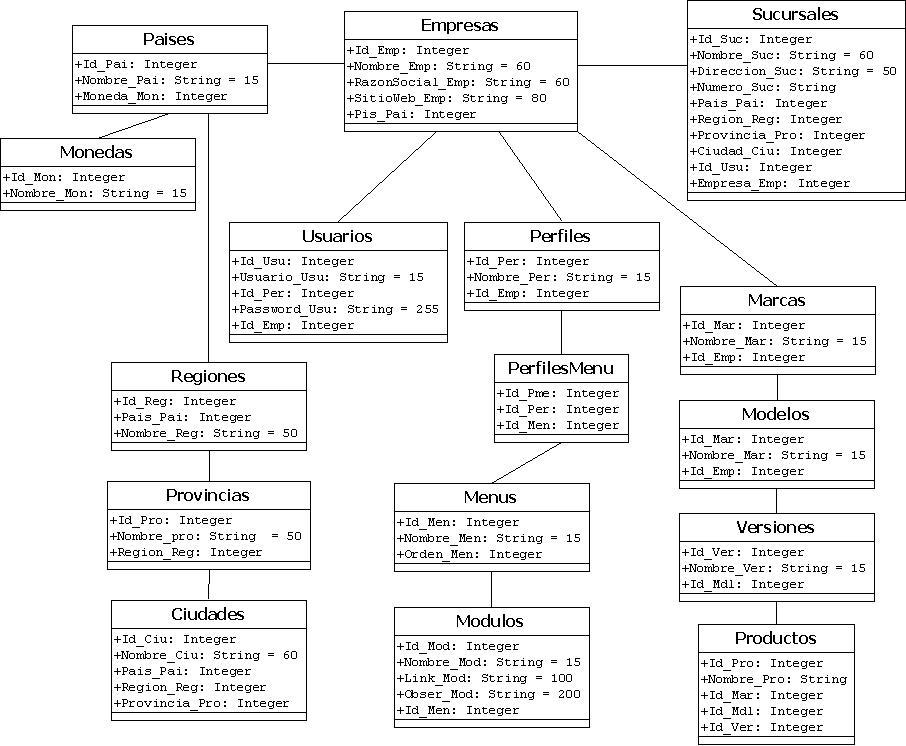
*[En UML el diagrama de clases es uno de los tipos de diagramas o símbolo estático y tiene como fin describir la estructura de un sistema mostrando sus clases, atributos y relaciones entre ellos. Estos diagramas son utilizados durante el proceso de análisis y diseño de los sistemas informáticos, en donde se intentan conformar el diagrama conceptual de la información que se manejará en el sistema. Como ya sabemos UML es un modelado de sistema Orientados a Objetos, por ende los conceptos de este paradigma se incorporan a este lenguaje de modelado.*

*Los diagramas de clases tienen las siguientes características:*

* *Las clases define el ámbito de definición de un conjunto de objetos.*
* *Cada objeto pertenece a una clase.*
* *Los objetos se crean por instanciación de las clases.*

*En su representación gráfica contamos con:*

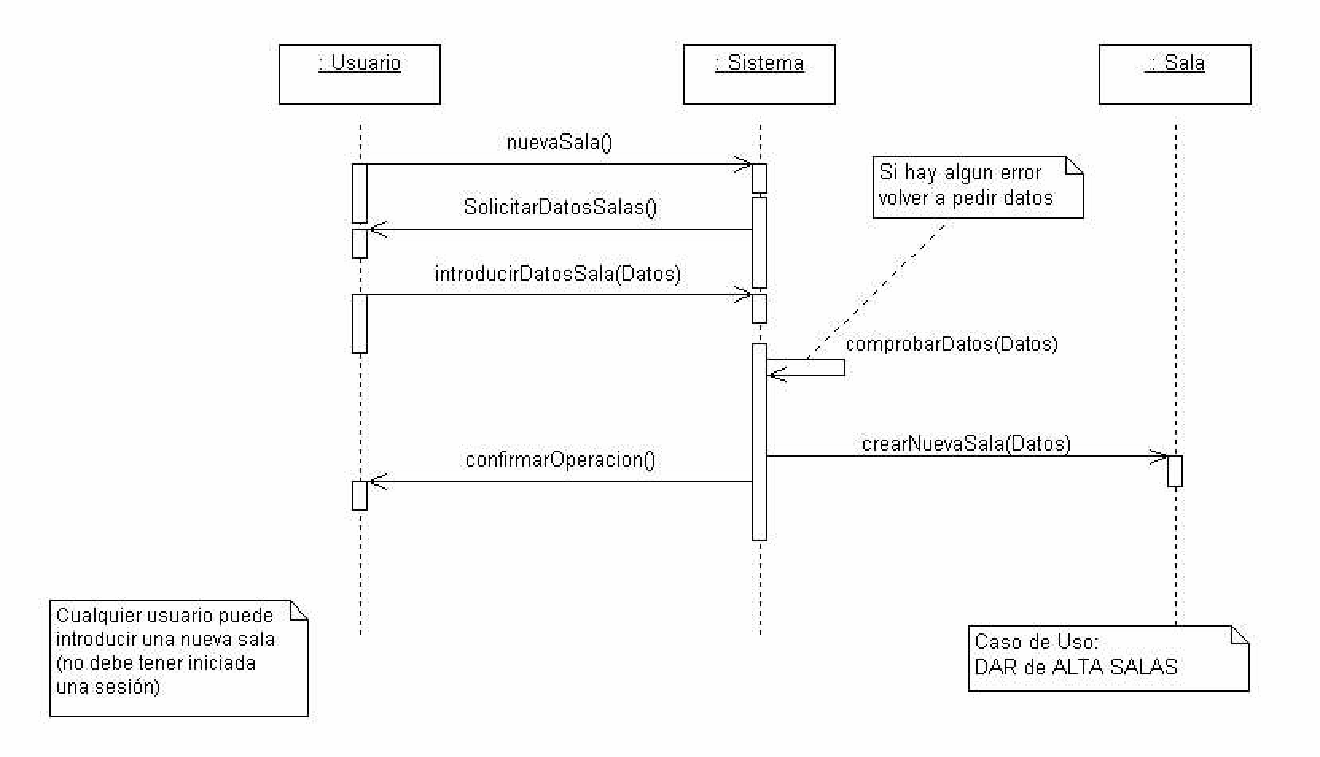
* *Nombre de la Clase.*
* *Atributos de la Clase.*
* *Operaciones con las Clases. ]*

**

**Diagrama de Secuencia**

*[En el diagrama de secuencia se muestra la interacción de los objetos que componen un sistema de forma temporal.*

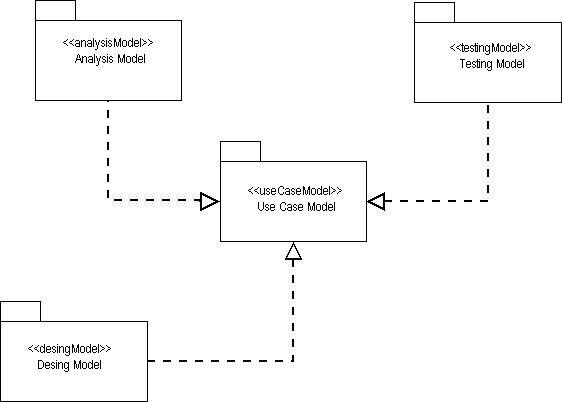
*Un diagrama de secuencia muestra la interacción de un conjunto de objetos en una aplicación a través del tiempo y se modela para cada método de la clase]*

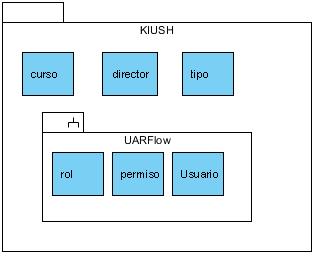


**Diagramas de Paquetes**

*[Los diagramas de Paquetes se usan para reflejar la organización de paquetes y sus elementos. Los usos más comunes de para los diagrama de paquete son para organizar diagramas de casos de uso y diagramas de clases, estos paquetes son como grandes contenedores de clases.*

*Los elementos contenidos en un paquete comparten el mismo espacio de nombres, esto significa que los elementos contenidos en un mismo espacio de nombres específico deben tener nombres únicos. Como otra característica de estos diagramas, cada paquete se debe identificar con un nombre único y opcionalmente mostrar todos los elementos dentro del mismo. ]*





**Diagrama de Colaboración**

*[Un diagrama de colaboración, se puede decir que es una forma alternativa al diagrama de secuencias a la hora de mostrar un escenario.*

*Este tipo de diagrama muestra las interacciones que ocurren entre los objetos que participan en una situación determinada. A diferencia del diagrama de secuencia, el diagrama de colaboración se enfoca en la relación entre los objetos y su topología de comunicación. En estos diagramas los mensajes enviados de un objeto a otro se representa mediante flechas, acompañado del nombre del mensaje, los parámetros y la secuencia del mensaje.*

*Estos diagramas están indicados para mostrar una situación o flujo de programa específico y son considerados uno de los mejores diagramas para mostrar o explicar rápidamente un proceso dentro de la lógica del programa. ]*

