Sistema Antiplagium

Documento de Estándares de Diseño

Versión 1.0

Historial de Revisión

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 08/04/2010 | 1.0 | Versión Inicial | Kim Alvarado  Patricia Natividad |
|  |  |  |  |
|  |  |  |  |

Índice

[1. Objetivos 4](#_Toc265232564)

[2. Estándares 4](#_Toc265232565)

[**2.1.** **Diagrama de clases** 4](#_Toc265232566)

[**2.2.** **Diccionario de datos** 5](#_Toc265232567)

[**2.3.** **Diagramas de secuencias** 6](#_Toc265232568)

[**2.4.** **Diagrama de componentes** 8](#_Toc265232569)

[**2.5.** **Diagrama de despliegue** 10](#_Toc265232570)

[**2.6.** **Diseño de Bases de Datos** 11](#_Toc265232571)

Estándares de diseño

1. **Objetivos**

El propósito de este documento es definir estándares para la elaboración del diseño del sistema Antiplagium.

De esta forma se logrará un mejor entendimiento, orden, y disminuir el riesgo de ambigüedades al detallar aspectos del diseño como algoritmos, pseudocódigo, especificaciones de clases, métodos y atributos, Diagrama de secuencias, Diagrama de estados, Diagrama de componentes y Arquitectura de Software.

1. **Estándares** 
   1. **Diagrama de clases**

El diagrama de clases permite visualizar las entidades físicas y abstractas representadas en el diseño del sistema, además de las relaciones y dependencias existentes entre ellas.

Una clase agrupa un conjunto de objetos que contienen atributos y sus acciones o métodos.

* **Estándares para nombrar una clase, sus atributos, métodos y relaciones:**
  + **Nombres de las clases:**
* El nombre de la clase debe iniciar con letra mayúscula.
* Si el nombre de la clase contiene más de una palabra se juntarán sin dejar espacio en blanco, el inicio de cada palabra se escribirá con letra Mayúscula.
  + **Nombres de los atributos:**
* El nombre de los atributos debe iniciar con letra minúscula.
* Si el nombre consta de más de una palabra colocarlas juntas sin dejar espacio, al inicio de la segunda y posteriores palabras colocarlas al inicio con letra mayúscula.
  + **Nombres de los métodos:**
* El nombre de los métodos sigue la misma sintaxis que los atributos.
  + **Asociaciones:**
* Las asociaciones deben indicar la cardinalidad en ambos sentidos.
* **Ejemplo de diagrama de clases:**

****

* 1. **Diccionario de datos**

El propósito del diccionario de datos es explicar detalladamente la información y significado de cada clase y atributos en el diagrama de clases.

* **Estándares para el diccionario de datos:**

Se seguirá el siguiente modelo para la realización de los diccionarios de datos:

|  |  |  |  |
| --- | --- | --- | --- |
| **<Nombre de Clase>** | | | |
| <Breve descripción de la clase> | | | |
| **Atributo** | **Tipo** | | **Descripción** |
| <Nombre del Atributo 1> | <Nombre Tipo 1> | | <Descripción del atributo 1> |
| <Nombre del Atributo 2> | <Nombre Tipo 1> | | <Descripción del atributo 2> |
| **Métodos** | | | |
| **<Nombre del Método 1>** | | | |
| **Devuelve:** <Tipo de valor devuelto> | | | |
| <Descripción del método 1> | | | |
| **Parámetro** | | **Tipo** | |
| <Parámetro 1> | | <tipo del parámetro 1> | |
| <Parámetro 2> | | <tipo del parámetro 2> | |
| **<Nombre del Método 2>** | | | |
| **Devuelve:** <Tipo de valor devuelto> | | | |
| <Descripción del método 2> | | | |
| **Parámetro** | | **Tipo** | |
| <Parámetro 1> | | <tipo del parámetro 1> | |
| <Parámetro 2> | | <tipo del parámetro 2> | |

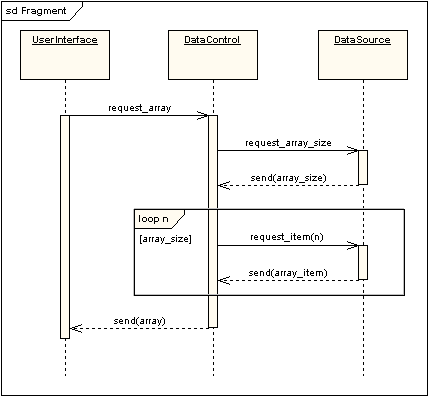
* 1. **Diagramas de secuencias**

Los diagramas de secuencia sirven para mostrar la interacción que se realiza entre los objetos a lo largo del tiempo en un determinado caso de uso. Dichas interacciones son representadas como mensajes dibujados como flechas de la línea de vida de origen hasta la línea de vida de destino.

* **Para los mensajes:**
* Los mensajes están formados por un nombre y sus respectivos parámetros.
* Se empleará notas para describir la información más detallada.
* La notación para los mensajes será la siguiente:

|  |  |  |
| --- | --- | --- |
| **Representación** | **Tipo de Mensaje** | **Utilizado para** |
|  | Simple | Para transferir el control de un objeto a otro. |
|  | Sincrónico | Para indicar que el objeto que emitió el mensaje esperará una respuesta para poder continuar con su trabajo |
|  | Asincrónico | Para indicar que el objeto que emitió el mensaje no esperará una respuesta para poder continuar con su trabajo |
|  | De Retorno o Respuesta | Para indicar el mensaje de retorno o respuesta. |

* **Para los objetos:**
* Se empleará cuadros de activación para describir el tiempo que durará un objeto para terminar una determinada tarea.
* Cuando se desea destruir un objeto se representará con un aspa “X” al final de la línea de vida.
* **Adicionalmente:**
* Los diagramas de secuencia se realizarán de izquierda a derecha y de arriba hacia abajo para facilitar la lectura del diagrama.
* La línea vertical del diagrama representa el tiempo de vida del objeto y la dimensión horizontal muestra la disposición de los objetos.
* **Ejemplo de diagrama de secuencias:**



* 1. **Diagrama de componentes**

El diagrama de componentes permite representar el diseño de la arquitectura del sistema a un alto nivel.

La simbología utilizada para estos diagramas será:

|  |  |
| --- | --- |
| **Símbolo** | **Significado** |
|  | **Paquete**, sirve para agrupar componentes o interfaces. |
|  | **Componente,** parte física y reemplazable del sistema. Agrupacion de clases e interfaces. |
|  | **Interface,** conjunto de métodos que permite la interacción entre componentes. |
|  | **Dependencia**, relación de dependencia entre componentes. |
|  | **Comentario**, para colocar anotaciones en el diagrama de componentes cuando sea necesario. |

* **Ejemplo de diagrama de componentes**



* 1. **Diagrama de despliegue**

Los diagramas de despliegue son utilizados para mostrar gráficamente la implantación del sistema, representando los dispositivos de hardware por medio de nodos.

La simbología a utilizar será:

|  |  |
| --- | --- |
| **Símbolo** | **Significado** |
|  | **Nodo**, dispositivo de hardware (Servidor, PC, etc.) |
| ------------------------- | **Comunicación**, relación entre nodos que implica comunicación directa. |

Además será necesario indicar los componentes que pertenecen a cada nodo con la siguiente tabla:

|  |  |
| --- | --- |
| Nodo | Componentes |
| Nombre del nodo 1 | Nombre Componente 1  Nombre Componente 2 |
| Nombre del nodo 2 | Nombre Componente 3  Nombre Componente 4 |

* Ejemplo de diagrama de despliegue



* 1. **Diseño de Bases de Datos**

A continuación se detallan las consideraciones para el diseño de tablas y campos en las bases de datos implicadas en el sistema Antiplagium.

* **Tablas**
  + **Nombres de las tablas:**
  + El nombre de la tabla debe estar escrito completamente en letras mayúsculas.
  + Los nombres de las tablas serán especificados en singular en todos los casos.
  + Si el nombre de la tabla contiene más de una palabra se juntarán sin dejar espacio en blanco ni caracteres en medio.
  + Los nombres de las tablas que sean asociaciones de otras tablas se formarán por el nombre de las dos tablas asociadas separados por la letra “X”. **Ejm**: PEDIDOXCLIENTE.
  + **Nombres de los campos:**
  + El nombre de los campos de las tablas deben iniciar con letra minúscula.
  + Si el nombre consta de más de una palabra colocarlas juntas sin dejar espacio, al inicio de la segunda y posteriores palabras colocarlas al inicio con letra mayúscula.
  + **Caracteres especiales:**
  + La letra “ñ” será reemplazada por “nh”.
  + Las letras tildadas se reemplazarán por su equivalente sin tilde.
  + No se utilizarán caracteres no alfanuméricos como puntos o guiones.
  + **Llaves primarias**
  + Los campos que sean llave primaria de una tabla deberán ser nombrados como “id”+ nombre de la tabla. **Ejm**: Llave de la tabla Usuario = idUsuario.
  + Los campos que sean llave primaria de una tabla deberán ser siempre los primeros en ser especificados (deben ir al inicio).
  + Los campos que sean llave foránea deberán ser nombrados de la misma manera que las llaves primarias.

