MistiStore

Plan de Arquitectura de Software

Versión 0.1

Historial de Revisiones

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| 11/06/12 | 0.1 | Traducción de títulos. Versión preliminar del documento. | Eddy Cáceres Huacarpuma |

Tabla de contenidos

1.- Introducción 4

1.1- Propósito 4

1.2- Alcance 4

1.3.- Definiciones, Acrónimos y Abreviaturas 4

2.- Objetivos y restricciones de la arquitectura 4

3.- Vista Lógica 5

3.1- Diagrama de Clases de Diseño 5

Plan de Arquitectura de Software

# 1.- Introducción

## 1.1- Propósito

El presente documento provee una visión inicial para la arquitectura de la aplicación Tienda Virtual MistiStore. De esta manera, se busca capturar y asentar las decisiones importantes que serán tomadas en el desarrollo de la aplicación.

## 1.2- Alcance

Se muestra a alto nivel el diseño de la arquitectura por vistas de la aplicación. En cada una, se presentan los diagramas correspondientes, a saber: modelo conceptual, diagrama de clases, casos de uso, diagramas de interacción, entre otros.

## 1.3.- Definiciones, Acrónimos y Abreviaturas

ARQUITECTURA DE SOFTWARE: conjunto de elementos estáticos, propios del diseño intelectual del sistema, que definen y dan forma tanto al código fuente, como al comportamiento del software en tiempo de ejecución. Naturalmente este diseño arquitectónico ha de ajustarse a las necesidades y requisitos del proyecto.

VISTAS: es una representación de un área de interés o perspectiva del sistema en alto nivel.

STAKEHOLDER: Individuo, equipo u organización con intereses relativos al sistema.

MODULO O COMPONENTE: cualquier elemento estructural abstracto, visible, externo, de alto nivel, analizable, que pueda constituir una funcionalidad de la solución del sistema.

# 2.- Objetivos y restricciones de la arquitectura

El desarrollo de la aplicación se enfoca en que llegue a tener características que sean sostenibles, eficientes y fáciles de usar para cualquier usuario del sistema.

Nuestras principales metas a nivel de arquitectura son las siguientes:

• Performance: El desempeño de la aplicación debe ser muy eficiente de tal manera el usuario inmediato y todos los demás observen lo más rápidamente posible los cambios realizados en un momento determinado.

• Usabilidad: El diseño debe ser orientado por y para la comodidad del usuario, de manera que la interfaz sea intuitiva y fácil de manejar, al mismo tiempo que se fomente altamente la interacción entre ambos. De la misma forma, el usuario debe tener la capacidad de equivocarse y regresar a un estado seguro en el que se le permita cumplir con su objetivo original sin que se le haga tedioso o complicado el proceso para llegar a dicho fin.

En la planificación de este proceso hemos encontrado las siguientes restricciones:

1. Restricciones de contenido: debido a que nuestro sistema está basado y propuesto inicialmente para el mercado peruano, el mismo debe estar de acuerdo al marco legal vigente comercial.
2. Restricciones de tecnología y uso de herramientas de desarrollo: están predefinidos los instrumentos a utilizar así como también la plataforma tecnológica sobre la que se va a desarrollar el sistema. La aplicación será implementada usando JSP, es por ello que las herramientas utilizadas estarán determinadas por las funcionalidades ofrecidas por dicho lenguaje.

# 3.- Vista Lógica

## 3.1- Diagrama de Clases de Diseño

