# **1.** **Introducción**

## **1.1** **Propósito**

El documento de arquitectura de software (DAS) muestra la arquitectura que se usará para implementar el sistema del Proceso de almacenamiento a través de la vista arquitectónica 4+1, cabe resaltar que la vista de casos de uso se ha detallado en otro documento. Este documento busca brindar una visión general y entendible del diseño propuesto para este sistema de almacén.

## **1.2** **Alcance**

Este documento explicará el desarrollo de la vista lógica, de procesos, de implantación, de implementación y de datos. Se estará considerando la vista de procesos debido a que el sistema de almacén tiene alta concurrencia y la vista de datos debido a que se usará una aplicación de base de datos externa.

## **1.3** **Definiciones, Siglas, y Abreviaciones**

[Esta subdivisión proporciona las definiciones de todos los términos, las siglas, y las abreviaciones necesarias para interpretar apropiadamente el **DAS**. Esta información puede ser proporcionada a través de la referencia al Glosario del Proyecto.]

## **1.4** **Referencias**

| **Documento** | **Versión** | **Fecha de la versión** |
| --- | --- | --- |
| Documento ERS | 1.1 | 4/ Jul/ 08 |
| Lista de Riesgos | 1.1 | 21/ Oct/ 08 |
| Plan de Iteración | 1.1 | 21/ Oct/ 08 |
| Documento de visión del Sistema | 1.1 | 4 / Jun/ 08 |

## **1.5** **Vista Global**

[Esta subdivisión describe lo que contiene el resto del **DAS** y explica cómo el documento está organizado.]

# **2.** **Representación Arquitectónica**

Se indica una descripción de lo que es la arquitectura del software para el sistema actual y cómo es representado. Se enumeran las vistas que son necesarias y se explica, para cada una, los tipos de elementos de modelaje que contiene.

**Vista de casos de uso:**

Son la lista de los casos de uso o escenarios del modelo de casos de uso que representan funcionalidades del sistema final.

**Vista lógica**

Describe las partes arquitectónicamente significativas del modelo de diseño, como ser la descomposición en capas de subclases, subsistemas o paquetes. Una vez presentadas estas unidades lógicas principales, se profundiza en ellas hasta el nivel que se considere adecuado.

**Vista de procesos:**

**Vista de Deployment:**

Describe uno o más escenarios del ambiente en el que será instalado y ejecutado el sistema. Muestra la comunicación entre los diferentes nodos que componen los escenarios antes mencionados.

**Vista de implementación:**

**Vista de datos:**

# **3.** **Metas y Restricciones Arquitectónicas**

La meta principal de la arquitectura del sistema será mostrar las características principales del sistema, teniendo en cuenta las siguientes metas y restricciones.

3.1.Metas

1.El Sistema de Almacenamiento permitirá a los usuarios acceder al sistema desde cualquier ordenador

.

2.Para poder acceder al Sistema de Almacenamiento será necesario el uso de un usuario y contraseña válidos, dependiendo del tipo de usuario se habilitarán distintas opciones.

3.Los requerimientos estipulados en la visión deberán ser implementados

.

4.El uso del sistema solo será posible si es que se encuentran dentro del almacén

3.2.Restricciones del Sistema

1.El sistema usará SQL como motor de la base de datos y se contará indispensablemente con una copia de respaldo en caso ocurra cualquier error

2.Las computadoras usadas para que puedan ejecutar el sistema de almacenamiento deberán tener como mínimo 1 GB de RAM y 500 MB de espacio libre en el disco duro y un SO windows en cualquiera de sus versiones

3.Las computadoras que utilicen el sistema de almacenamiento deberán de tener instalado el JRE 1.6 o superior ya que el sistema será desarrollado en Java

# **4.** **Vista de Casos de Uso**

Ver el documento ERS referenciado en la sección 1.4 Referencias

# **5.** **Vista Lógica**

[Se describen las partes del modelo de diseño que son significativas arquitectónicamente, tales como, su descomposición dentro de subsistemas y paquetes de servicios; y para cada paquete, su descomposición dentro de clases y utilidades de clases. Introducir las clases relevantes arquitectónicamente y describir sus responsabilidades, así como sus relaciones, operaciones y atributos.]

## **5.1** **Visión general**

[Describa de manera general, la descomposición del modelo de diseño en términos de su jerarquía de paquetes y capas. Incluir algunos diagramas mostrando los paquetes de alto nivel, así como su interdependencia y suscapa]s.

## **5.2** **Paquetes de Diseño Significativos Arquitectónicamente**

[Para cada paquete significativo, incluya una subsección con su nombre, una breve descripción y un diagrama con todas las clases y paquetes significativos, contenidos dentro del paquete. Para cada clase significativa en el paquete, incluya su nombre, una breve descripción y, opcionalmente, alguna descripción de sus principales responsabilidades, operaciones y atributos.]

## **5.3** **Realizaciones de los casos de uso**

[Esta sección ilustra cómo el software realmente trabaja, al mostrar la realización de algunos casos de uso seleccionados (o escenarios), y explica cómo los diferentes elementos del modelo de diseño contribuyen a su funcionalidad]

## **5.4** **Diagrama Conceptual**

# **6.** **Vista de Procesos**

# *[Se describe la descomposición del sistema dentro de procesos y grupos de procesos. Se debe organizar la sección por grupos de procesos donde se reflejen sus comunicaciones o interacciones. Describa primero los principales modos de comunicación entre procesos, tales como: pases de mensajes, etc. Para cada red de procesos se debe incluir una subsección con la siguiente información:*

- *Su nombre.*

- *Los procesos involucrados.*

- *Las interacciones entre los procesos en la forma de diagramas de colaboración, en los cuales los objetos son los procesos reales que completan su propio vínculo de control. Para cada proceso, describa brevemente su comportamiento, su ciclo de vida y sus características de comunicación.]*

# **7.** **Vista de Implantación**

# **[***Se describe una o más configuraciones de redes físicas (hardware) sobre las cuales el software será desarrollado. Para cada configuración, se debe indicar los nodos físicos (computadores, CPUs) que ejecutan el software, y sus interconexiones (bus, LAN, punto a punto, etc.). También se incluye un mapa de los procesos de la vista de procesos, dentro de los nodos físicos. Para cada configuración física de red incluya una subsección con la siguiente información:*

- *Su nombre.*

- *Un diagrama de entrega que ilustre la configuración, seguida por un mapa de procesos para cada procesador.*

# **8.** **Vista de Implementación**

[Se describe la descomposición del software dentro de capas y subsistemas en el modelo de implementación].

Contiene dos subsecciones:

## **8.1** **Vista General**

En este sistema utilizaremos el Modelo Vista Controlador (MVC), el cual es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos. En nuestro sistema la vista es la página Web, que será desarrollada en PHP y el código que provee de datos dinámicos a la página, el modelo es el Sistema de Gestión de Base de Datos y el controlador representa la Lógica de negocio.

## **8.2** **Capas**

*[Para cada capa, incluya una subsección con la siguiente información:*

- *Su nombre.*

- *Una lista de los subsistemas ubicados en la capa, con nombre, abreviación y una breve descripción.*

- *Un diagrama de componentes que muestre los subsistemas y sus dependencias importantes.*

- *Su relación con elementos de la Vista Lógica.]*

# **9.** **Vista de Datos**

[Una descripción de la perspectiva del almacenamiento de datos persistentes en el sistema. Esta vista es sólo opcional en el caso de Aplicaciones Adquiridas.]

# **10.** **Tamaño y Desempeño**

# *[Se describen las principales características de dimensionamiento del software que impactan la arquitectura, así como las restricciones de desempeño: rendimiento, tiempos de respuesta, etc.]*

# **11.** **Calidad**

# *[Se describe cómo la arquitectura del software contribuye con todas las capacidades (además de la funcionalidad) del sistema: extensibilidad, confiabilidad, portabilidad, etc. Esta sección puede ser organizada por vistas.]*