Documento de arquitectura de software

SUSO 1121-2023

Control del documento

Proyecto

Sistema de gestión de inventario y reportes

Titulo

Arquitectura del sistema – [V1.0 al 30 de abril de 2023]

Generado por

Database administratos & Data analist – [Omar Caballero Meza]

Aprobado por

Artadi Ruiz Gonzalo

⦁ Introducción

⦁ Propósito

El siguiente documento proporciona una descripción de la arquitectura del sistema que se pretende implementar para gestionar el proceso de inventario y reportes para una empresa que realice ventas de productos fisicos. Utilizando diferentes tipos de vistas para representar un mismo sistema y las funcionalidades que hay dentro del mismo, se pretende capturar y transmitir las decisiones de arquitectura más significativas que se tomaron para crearlo.

⦁ Alcance

El documento contiene la representación del proyecto (SUPER SHOP), mediante el diseño de diagramas dado el análisis previo de los requisitos del sistema para darle solución a una problemática presentado por un cliente.

El documento está organizado mediante tres ideas principales:

⦁ Características generales del diseño.

⦁ Los requisitos funcionales y no funcionales para el diseño.

⦁ Los modelos y vistas que lo representan.

Los modelos implementados para la representación del sistema, permiten analizar el funcionamiento del sistema desde distintos puntos de vista, de forma que podemos implementar el “Modelo de arquitectura 4+1” de Krutchen. Gracias a esto, se puede llegar implementar el análisis de los requisitos, la implementación del diseño de solución, así como para la especificación, construcción y despliegue del sistema en su ambiente de explotación.

Cada uno de los modelos fueron implementados en la herramienta de modelado UML (Lenguaje de modelado unificado).

⦁ Usuarios interesados (StakeHolders)

El Documento de Arquitectura de Software (DAS), contribuye a la fácil comprensión para todos aquellos usuarios que deseen comprender el diseño y la arquitectura correspondiente al proyecto.

⦁ Recomendaciones de conformidad con esta práctica.

N/A

⦁ Referencias

Para llevar a cabo la realización de este documento, se basó la información en las siguientes referencias:

⦁ IEEE 830-1998 ST

⦁ ISO 9126 – 2001 Calidad de software y métricas de evaluación.

⦁ The 4+1 View. Krutchen.

⦁ Definiciones, acrónimos y abreviaciones

DAS: Documento de arquitectura de software.

SUPER SHOP: Sistema de Gestión de inventario con generador de reportes para indistintas empresas

HTTP: Protocolo para la transferencia de hipertextos.

TCP: Protocolo para el control de transmisión.

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.

DESCRIPCIÓN DE ARQUITECTURA: Colección de productos de documentación.

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

TIPOS DE VISTAS: Especificación de una convención de cómo construir y usar una vista. Deben satisfacer la capacidad de creación y análisis de una vista.

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

ESCENARIO: Especifica el comportamiento y limita el interés de un área específica del sistema para uno o varios stakeholders.

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

ATRIBUTOS DE CALIDAD: Un atributo de calidad, es una cualidad deseable de la solución, que pueda manifestarse en forma de requerimiento no funcional, que pueda ser medible, testeable y finalmente evaluable.

⦁ Framework conceptual

⦁ Descripción de la arquitectura en contexto

El documento representa la arquitectura del software mediante la implementación de diferentes vistas relacionadas al “Modelo de vistas 4+1” de Kruchten. Cada una de estas vistas, representan el sistema desde distintos puntos de vista para los interesados dentro del proyecto. Las vistas son las siguientes:

⦁ Vista lógica.

⦁ Vista física.

⦁ Vista de desarrollo.

⦁ Vista de procesos.

Aparte existe la vista adicional que reúne los atributos de cada una de las vistas anteriores, y las plasma en una vista de escenarios. Cada una de las vistas pertenecientes a este modelo poseen su representación a través de modelos realizados en la herramienta StarUML.

⦁ Stakeholders y sus roles

El presente documento identifica las funciones de los interesados o mejor conocidos como stakeholders a partir de la interpretación de los casos de uso del sistema.

4.3 Actividades de arquitectura en el ciclo de vida

N/A.

4.4 Usos de las descripciones de arquitectura

Las descripciones de arquitectura de este documento se usarán para referenciar el diseño del sistema de software de SIGFOP.

5. Descripciones prácticas de arquitectura

N/A.

5.1. Documentación de la arquitectura

N/A.

5.2. Identificación de los Stakeholders y sus responsabilidades

Stakeholder Descripción Escenario

Cliente Persona que interactúa con el sistema para solicitar estudios fotográficos o productos. ⦁ Escenario de negocio.

⦁ Escenario de diseño.

Administrador Persona encargada de gestionar cada uno de los procesos ejecutados dentro del sistema. ⦁ Escenario de negocio.

⦁ Escenario de diseño.

Fotógrafo Persona encargada de efectuar los estudios fotográficos solicitados por el cliente. ⦁ Escenario de negocio.

⦁ Escenario de diseño.

Diseñador Persona encargada de editar los estudios fotográficos de acuerdo a las necesidades del cliente. ⦁ Escenario de negocio.

⦁ Escenario de diseño.

Impresor Persona encargada de gestionar la impresión de cada uno de los estudios fotográficos. ⦁ Escenario de negocio.

⦁ Escenario de diseño.

Atención al cliente Persona encargada de interactuar con los clientes. ⦁ Escenario de negocio.

⦁ Escenario de diseño.

5.3. Selección de puntos de vista de la arquitectura

Vistas UML

Escenarios Casos de uso

Lógica Clases

Desarrollo Componentes

Física Despliegue

Procesos Secuencia

5.4. Vistas de arquitectura

Vistas - Escenarios (+1)

Diagrama de casos de uso

Vistas – Vista lógica

Diagrama de clases

Diagrama de secuencias para la solicitud de estudios

Diagrama de secuencias para el proceso de un estudio fotográfico

Vistas – Vista de desarrollo

Diagrama de componentes general

Diagrama de paquetes

Vistas – Vista física

Modelo de despliegue

Vista – Vista de procesos

Diagrama de actividades

DERSCRIPCION DE CONECTORES

5.6 Arquitectura lógica.

Performances

La arquitectura de software escogida apoya a los requerimientos no funcionales y requerimientos de arquitectura de sistemas descritos en los anexos de este documento.

1. El sistema apoyará hasta 2000 usuarios simultáneos contra la base de datos central en cualquier tiempo dado, y hasta 500 usuarios simultáneos contra los servidores locales en un momento dado.

2. El sistema proporcionará el acceso a la base de datos de catálogo de curso de herencia sin más que una 10 segunda latencia.

3. El sistema debe ser capaz de completar el 80 % de todas las transacciones dentro de 2 minutos.

4. La parte de cliente requerirá el espacio de disco de menos de 20 MB y la RAM de 32 MB.

Calidad

La arquitectura de software apoya las exigencias de calidad, como estipulado en la especificación anexa a este documento.

1. El interfaz de usuario será LOCAL.

2. El interfaz de usuario del Sistema SUPERSHOP será diseñado para la facilidad de uso y será apropiado para asegurar las normas de usabilidad universal establecidas por ISO 9126.

3. Cada despliegue de opciones de pantalla, tendrá la ayuda en línea para el usuario. La ayuda En línea incluirá paso a paso instrucciones en la utilización del Sistema. La ayuda En línea incluirá definiciones para términos y acrónimos.

5.8 Detalles de la implementación

La especificación de un sistema intensivo en software tiene como última representación al código fuente de los componentes. Este código indica los más finos detalles del software, por medio de un lenguaje preciso, capaz de ser traducido automáticamente a instrucciones de la máquina. Acompaña al código, las llamadas previsiones de compilación, constituidos por todos los elementos de soporte necesarios para realizar la construcción de los componentes a partir del conjunto de códigos. Esta

sección detalla la obtención y uso del paquete de código fuente para el proyecto. De manera de facilitar el uso de este, para las futuras ampliaciones o correcciones del sistema.

5.8.1 Lenguajes y plataformas

La lógica de diseño arquitectónico aplicada en este documento, abre la posibilidad de que la implementación de bajo nivel sea efectuada con lenguajes que solamente cumpla con la característica de Orientación a Objetos (python). Y eso va a depender directamente de las características de los desarrolladores, capacidad de aprendizaje, y en muchos casos opciones propias de la empresa para la cual se efectúa el diseño. Si la implementación se desea desarrollar bajo lenguajes que no cumplan las características mencionadas, se deberá confeccionar una nueva vista que cumpla con los requerimientos funcionales y no funcionales de los stakeholders que lo solicitan.