**1. Propósito.**

Establecer los puntos que debe cubrir como referencia documental mínima un documento de Diseño de sistemas automatizados.

**2. Ámbito de responsabilidad.**

RDSI Responsable del Desarrollo de Sistemas de Información.

DIU Diseñador de interfaz de usuario.

ASI Arquitecto de Sistemas de Información.

**3. Definiciones.**

Referencia *Términos y Definiciones* en Modelo del Proceso para la Administración del Desarrollo de Sistemas de Información Automatizados.

**4. Método de Trabajo.**

|  |  |
| --- | --- |
| ***No.*** | ***Instrucción*** |
| 1 | **Criterios, Convenciones y recomendaciones para utilizar este instructivo**Los *elementos de configuración* elaborados con este instructivo deberán ser creados y actualizados tomando en consideración los siguientes datos específicos de este *elemento de configuración*:

|  |  |
| --- | --- |
| *TÍTULO*  | La leyenda DISEÑO, con letras mayúsculas y tipo de letra normal negrita. |

Los apartados que debe contener la tabla de contenido del documento que se elabore son lossiguientes:**0. Historia****1. Introducción****2. Diseño de interfaz gráfica de usuario**2.1 Vista principal2.2 Módulos  2.2.1 Descripción del módulo 2.2.2 Vista primaria 2.2.2.1 Propósito 2.2.2.2 Objetos y acciones 2.2.2.3 Dimensiones visuales 2.2.2.4 Ejemplo visual o screenshot 2.2.3 Vistas secundarias2.3 Estándares de gráficos2.4 Jerarquía de navegación**3. Diseño Arquitectónico**3.1 Representación de la Arquitectura3.2 Metas y Restricciones de la Arquitectura**4. Diseño de Casos de Uso**4.1 Identificador de Caso de Uso4.2 Diagramas de Interacción 4.2.1 Diagrama de Secuencia 4.2.2 Objetos Participantes**5. Vista Lógica**5.1 Diagrama de paquetes 5.2 Descripción de paquetes 5.3 Diagrama de clases 5.4 Descripción de clases 5.5 Esquema de la base de datosDespués de la última página de la *Tabla de Contenido* se deberá desarrollar el contenido de cada uno de los apartados incluidos.Los siguientes apartados de este instructivo de trabajo describen el contenido de los apartados correspondientes del documento en elaboración. |
| 2 | **Historia (0. Historia “Diseño”)**En este apartado se registran los eventos relevantes durante la elaboración del documento, tales como: creación, revisión, autorización, adición, modificación, actualización, etc. Para cada evento se debe registrar la fecha, el nombre del responsable y un breve comentario sobre la acción realizada. |
| **3** | **Introducción (1. Introducción “Diseño”)**En este apartado se plasma una breve reseña del proceso del usuario solicitante. Se describe la importancia, el alcance y se proponen los objetivos y lineamientos para la elaboración del documento de diseño. |
| **4** | **Diseño de Interfaz Gráfica de Usuario (2. Diseño de Interfaz Gráfica de Usuario “Diseño”)**La Interfaz Gráfica del Usuario es el elemento del Sistema de Información con el cual el usuario interactúa. Por este motivo es importante que sea funcional y de fácil uso. El diseño gráfico dependerá de la herramienta de desarrollo utilizada, pero independientemente de esto, al construir esta interfaz se deberán establecer directrices para el diseño de la interfaz de usuario, para la introducción de datos y para la visualización de la información. |
| **5** | **Modelo Funcional (2.1. Vista principal “Diseño”)**En este apartado se describe de forma general cómo estará conformada la pantalla principal del sistema y se especifican los diferentes módulos, menús u opciones que la componen.  |
| **6** | **Modelo Funcional (2.2. Módulos “Diseño”)**En este apartado se detalla para cada módulo, menú u opción en base a cómo estará compuesta la interfaz gráfica correspondiente. |
| **7** | **Modelo de Datos (2.2.1 Descripción del Módulo “Diseño”)**En este apartado se deberá describir la funcionalidad esperada para cada módulo descrito en el punto anterior al cual se le va a desarrollar su interfaz gráfica.  |
| **8** | **Vista Primaria (2.2.2 Vista primaria “Diseño”)**En esta sección se describen las partes que forman la vista general del módulo, identificando los componentes que conforman la vista, las dimensiones esperadas y se da un ejemplo visual. |
| **9** | **Propósito (2.2.2.1 Propósito** **“Diseño”)**En esta sección se describe en forma breve el objetivo principal del módulo en cuestión. |
| **10** | **Componente y acciones (2.2.2.2 Componentes y acciones “Diseño”)**En este apartado se deberá realizar una clara identificación de la estructura que tendrá la vista a desarrollar. Especificando los componentes que la conforman y las acciones que permiten realizar dichos componentes. Se recomienda mostrar dicha relación a manera de una tabla como en el siguiente ejemplo:

|  |  |
| --- | --- |
| **Componentes** | **Acciones** |
| Cuadro de texto - Número del usuario  | Permite al usuario introducir su número de usuario. |
| Cuadro de texto - Contraseña | Permite al usuario introducir su contraseña. |
| Botón - Aceptar | Lanza la validación de usuario y contraseña para permitir o denegar el acceso en base a las condiciones establecidas. |
| Botón - Cancelar | Permite al usuario eliminar el contenido de los campos de texto mencionados anteriormente. |

 |
| **11** | **Dimensiones visuales (2.2.2.3 Dimensiones visuales “Diseño”)**Las dimensiones visuales se refieren a los atributos de posición, tamaño, forma y color que se esperan tenga el módulo correspondiente.  |
| **12** | **Ejemplo visual (2.2.2.4 Ejemplo visual o screenshot “Diseño”)**Aquí se muestra a manera de ejemplo una base de cómo se pretende se vea la interfaz, la cual es sólo una guía para el diseño final de la interfaz del sistema y puede variar dependiendo de la herramienta final con que se desarrolle. |
| **13** | **Vistas secundarias (2.2.3 Vistas secundarias “Diseño”)**Se determina vista secundaria a todas aquellas pantallas o interfaces que se deriven de ejecutar alguna acción de la vista primaria y que se muestren, ya sea en una ventana independiente a la vista primaria o como parte del flujo de pantallas necesarias para completar una acción. En esta sección se debe desarrollar para cada vista secundaria los mismos apartados que de la vista primaria y son: Propósito, Objetos y acciones, Dimensiones visuales y Ejemplo visual o screenshot. |
| **14** | **Estándares de gráficos (2.3 Estándares de gráficos “Diseño”)**En este apartado se deberá definir la especificación detallada de los estándares que se deben seguir para la interfaz gráfica como son: esquemas de color, fuentes de escritura, formatos aceptados para imágenes (.GIF, .JPG, etc.) o alguna otra consideración importante para el sistema en cuestión. Nota: Es necesario tomar en cuenta la imagen institucional. |
| **15** | **Jerarquía de navegación (2.4 Jerarquía de navegación “Diseño”)**Aquí se pretende mostrar en un diagrama como es que se relacionan las diferentes pantallas o ventanas que componen al sistema. Para realizar esta sección del documento se recomienda generar una caja por cada módulo identificado en la sección “2.2 Módulos” de este documento. Y dentro de cada caja, incluir las diferentes vistas, tanto primarias como secundarias, que lo componen. Posteriormente, relacionar dentro de cada módulo las vistas por medio de flechas para indicar el flujo. Por último, es necesario indicar el flujo global, relacionando los módulos entre ellos por flechas. Para mayor referencia se muestra el siguiente ejemplo. |
| **16** | **Diseño Arquitectónico (3. Diseño Arquitectónico “Diseño”)**Es la estructura de programas que observará el sistema, resultado de una serie de refinamientos del modelo funcional basados en el análisis del flujo de transformación y transacción de datos. Es una descripción de la estructura modular del sistema y sus interfaces.El objetivo del diseño arquitectónico es desarrollar una estructura de programa modular y representar las relaciones de intercambio de información entre los módulos. El diseño arquitectónico deberá combinar la estructura del programa con las estructuras de datos, definiendo interfaces que faciliten el flujo de datos a través del programa.Cada elemento de primer nivel del modelo funcional deberá ser traducido en un módulo del diseño arquitectónico. Cada flujo de datos se deberá traducir en una línea que une a los módulos.  |
| **17** | **Representación de la Arquitectura (3.1 Representación de la Arquitectura “Diseño”)**En esta sección se detalla cómo es que se pretende mostrar la arquitectura de la aplicación, si es por medio de una serie de vistas de UML, si se utilizara un Modelo de Datos o un diseño procedimental.  |
| **18** | **Metas y Restricciones de la Arquitectura (3.2 Metas y Restricciones de la Arquitectura “Diseño”)**Se definen las metas que se pretenden lograr con la arquitectura propuesta, por ejemplo: si se trata de generar un sistema Web, o si se pretende tener una herramienta stand-alone, o si se realizará algún mantenimiento menor. De igual forma, con base en las metas, se detallan aquellas restricciones propias del sistema a desarrollar, por ejemplo: Sólo se podrá accesar al sistema Web por medio de Intranet o los componentes desarrollados deber poderse ejecutar en un servidor Jboss. |
| **19** | **Diseño de Casos de Uso (4. Diseño de Casos de Uso “Diseño”)**Esta sección es importante ya que representa aquellos casos de uso que son relevantes para la funcionalidad central del sistema. Describiendo los casos de uso que tienen una cobertura sustancial o que soportarán a la arquitectura cuando está sea sometida a estrés. Aquí se recomienda utilizar la notación UML para elaborar los diagramas necesarios. |
| **20** | **Identificador de Caso de Uso (4.1 Identificador de Caso de Uso “Diseño”)**En este apartado únicamente se incluye el nombre o identificador del caso de uso a diseñar. |
| **21** | **Diagramas de Interacción (4.2 Diagramas de Interacción “Diseño”)**Los diagramas de interacción, representan la forma en cómo un Cliente (Actor) u Objetos (Clases) se comunican entre sí en petición a un evento. Esto implica recorrer toda la secuencia de llamadas, de donde se obtienen las responsabilidades claramente.En los puntos que integran este apartado se incluye información relevante útil para la comprensión de esta sección. |
| **22** | **Diagrama de Secuencia (4.2.1 Diagrama de Secuencia “Diseño”)**Los Diagramas de Secuencia muestran las interacciones expresadas en función de secuencias temporales. En este apartado se desarrollan el o los diagramas correspondientes a cada caso de uso. |
| **23** | **Diagrama de Colaboración (4.2.2 Objetos Participantes “Diseño”)**En este apartado se enlistan y describen los objetos que participan en los diagramas de interacción realizados para cada caso de uso. |
| **24** | **Vista Lógica (5. Vista Lógica “Diseño”)**En este apartado se enlistan y describen los objetos que participan en los diagramas de interacción realizados para cada caso de uso. |
| **25** | **Diagrama de paquetes (5.1 Diagrama de paquetes “Diseño”)**Se muestra de forma gráfica como están generados los paquetes que componen el sistema y la relación que pueda existir entre ellos. Para cada paquete que contenga sub-paquetes se tiene que generar otro diagrama de paquetes. Para mayor referencia se presenta un ejemplo:  |
| **26** | **Descripción de paquetes (5.2 Descripción de paquetes “Diseño”)**Para cada paquete definido en el punto anterior de este documento, se da un detalle del mismo. Incluyendo datos como Descripción, Clases o programas correspondientes, Relaciones que posee el paquete con otros paquetes, y sub-paquetes incluidos.  |
| **27** | **Diagrama de clases (5.3 Diagrama de clases “Diseño”)**Un diagrama de clases sirve para visualizar las relaciones entre las clases que involucran el sistema las cuales pueden ser asociativas, de herencia, de uso y de contenimiento. El diagrama de clases incluye la relación entre un objeto y otro, la herencia de propiedades de otro objeto y los conjuntos de operaciones/propiedades que son implementadas para una interfaz. En esta sección se debe mostrar el diagrama de clases a utilizar para realizar el sistema.DG.png |
| **28** | **Descripción de clases (5.4 Descripción de clases “Diseño”)**En este apartado se describe cada una de las clases identificadas en el diagrama del punto anterior de este documento. Para cada clase se debe especificar lo siguiente:* Nombre: Identificador único de la clase.
* Descripción: Mostrar brevemente cual es la función de la clase.
* Responsabilidades: Este atributo se llena sólo cuando la clase tiene una responsabilidad específica que es importante mencionar. Por ejemplo: Asegura la importación de los datos desde un archivo de Excel.
* Relaciones: Aquí se indican las clases con las cuales se relaciona y el tipo de relación.
* Métodos: En este atributo se enlistan los métodos y para cada uno, se da una breve explicación de su función.
* Atributos: Aquí se enlistan los atributos que tiene la clase y para cada uno se especifica la información que va a contener.
 |
| **29** | **Esquema de la base de datos (5.5 Esquema de la base de datos “Diseño”)**En este apartado se muestra el diagrama Entidad-Relación que se va a utilizar para almacenar los datos necesarios para el sistema de información a desarrollar.  |

**5. Anexos.**

**No Aplica.**