**Guía de Diseño de Interfaz de Usuario**

**Sistema de Control de Entregas**

**Versión <1.2>**

**Revisión Histórica**

|  |  |  |  |
| --- | --- | --- | --- |
| **Fecha** | **Versión** | **Descripción** | **Autor** |
| <05/oct/15> | <1.0> | <Se agregó al documento la interfaz principal y las reglas de oro en el diseño de interfaces> | Michel Martínez |
| <28/dic/15> | <1.1> | Se agregaron los elementos RegularExpressionValidator  y RequiredFieldValidator | Borboa Lopez Jesus Bernardo, Michel Martínez, Alejandro López |
| <05/ene/16> | <1.2> | Se agregó dimensiones y colocación de elementos, y ubicación gráfica. | Borboa Lopez Jesus Bernardo, Michel Martínez, Alejandro López |
|  |  |  |  |
|  |  |  |  |

**Tabla de Contenido**

1. Introducción

2. Definición de Nombres de Elementos

2.1 Swing Components

2.2 Swing Containers

2.3 Swing Menus

3. Dimensiones y Colocación de Elementos

3.1 Swing Components

3.2 Swing Containers

3.3 Swing Menus

4. Ubicación de Elementos

4.1 Ubicación Gráfica

4.1.1 Captura de Mantenimiento a bitácora.

5. Reglas de Oro en el Diseño de Interfaces de Usuario

5.1 Darle al Usuario el Control

5.2 Reducir la carga de memoria al usuario

5.3 Hacer Consistente la Interface

**SISTEMA DE CONTROL DE ENTREGAS**

1. **Introducción**

El propósito de este documento es de definir los lineamientos que se deben seguir para la realización de las interfaces de usuario para el sistema SISTEMA DE CONTROL DE ENTREGAS. Esto con el fin de estandarizar la elaboración de éstas en cuanto al nombramiento y localización de los elementos que serán utilizados en las interfaces. La elaboración de estas interfaces también busca responder a las recomendaciones de las Reglas de Oro que permiten dar control al usuario sobre las mismas, reducir su carga de memoria y hacer consistente la presentación de las interfaces.

1. **Definición de Nombres de Elementos**
   1. **Swing Components**

|  |  |
| --- | --- |
| **Swing Components** | |
| **Elemento** | **Definición de nombre** |
| Button | Su nombre iniciará con **bt** y le seguirá el nombre o abreviatura de la acción que realiza.  Ej. btAceptar |
| Label | Su nombre iniciará con **et** y le seguirá el nombre o abreviatura de la información que se solicita en la caja de texto a la que acompaña.  Ej. Si la caja de texto requiere el nombre de la persona : etNombre |
| TextBox | Su nombre iniciará con **tx** y le seguirá el nombre o abreviatura de la información solicitada.  Ej. Se solicitan comentarios: txComentarios |
| DropDownList | Su nombre iniciará con **ddl** y le seguirá el nombre o abreviatura del tipo de opciones que contiene. |
| RequiredFieldValidator | Su nombre iniciara con **rfv** le seguirá el nombre o abreviatura de la información que se solicita en el elemento al que valida.  Ej. Si la caja de texto requiere el nombre de la persona : rfvNombre |
| RegularExpressionValidator | Su nombre iniciara con **rev** le seguirá el nombre o abreviatura de la información que se solicita en el elemento al que valida.  Ej. Si la caja de texto requiere el nombre de la persona : revNombre |
| anclas |  |
| Image | Su nombre iniciara con **img** yle seguirá el nombre o abreviatura de la imagen que muestra. |
| Gridview | Su nombre iniciará con gdv y le seguirá el nombre o abreviatura de la información a mostrar.  Ej. Se solicitan comentarios: gdvBitácora |
| GridViewRow | Su nombre iniciará con gdvr y le seguirá el nombre o abreviatura de la acción en la que se utilizará.  Ej. Se solicitan comentarios: gdvrConsultaEsp |
| Chart | Su nombre iniciara con **chart y** le seguirá el nombre o abreviatura de la información solicitada. |
| img | Su nombre iniciará con **img** yle seguirá el nombre o abreviatura de la imagen que muestra. |
| Login | Su nombre iniciará con **login** y le seguirá el nombre o abreviatura de la información solicitada. |

* 1. **Swing Containers**

|  |  |
| --- | --- |
| **Swing Containers** | |
| **Elemento** | **Definición del nombre** |
| Panel | Su nombre iniciará con **jp** y le seguirá el nombre o abreviatura de la información que contendrá. |
| HtmlForm | Su nombre iniciará con **form** y le seguirá el nombre o abreviatura de la información que contendrá. |
| header | Su nombre iniciará con el nombre o abreviatura de la información que contendrá. |
| div | Su nombre iniciará con el nombre o abreviatura de la información que contendrá. |
| section | Su nombre iniciará con el nombre o abreviatura de la información que contendrá. |
| nav | Su nombre iniciará con el nombre o abreviatura de la información que contendrá. |

* 1. **Swing Menus**

|  |  |
| --- | --- |
| **Swing Menus** | |
| **Elemento** | **Definición del nombre** |
| **LoginStatus** | Su nombre iniciará con **LoginStatus** y le seguirá el nombre o abreviatura de la información que contendrá. |

1. **Dimensiones y Colocación de Elementos**
   1. **Swing Components**

|  |  |
| --- | --- |
| **Swing Components** | |
| **Elemento** | **Definición de nombre** |
| Button | Sus dimensiones serán desde un ancho de 100px y una altura de 25 px dependiendo de la interfaz gráfica que se utilice. Su colocación dependerá de la interfaz gráfica que se utilice. |
| Label | Sus dimensiones serán desde un ancho de 30px y una altura de 30px dependiendo de la interfaz gráfica que se utilice. Su colocación dependerá de la interfaz gráfica que se utilice. |
| TextBox | Sus dimensiones serán desde un ancho de 30px y una altura de 30px dependiendo de la interfaz gráfica que se utilice. Su colocación dependerá de la interfaz gráfica que se utilice. |
| DropDownList | Sus dimensiones serán desde un ancho de 30px y una altura de 30px dependiendo de la interfaz gráfica que se utilice. Su colocación dependerá de la interfaz gráfica que se utilice. |
| anclas | Sus dimensiones serán desde un ancho de 30px y una altura de 30px dependiendo de la interfaz gráfica que se utilice. Su colocación dependerá de la interfaz gráfica que se utilice. |
| Image | Sus dimensiones serán desde un ancho de 40px y una altura desde 40px dependiendo de la interfaz gráfica que se utilice. Su colocación dependerá de la interfaz gráfica que se utilice. |
| Gridview | Sus dimensiones son dinámicas ya que depende la cantidad de información para asignarle una dimensión. Su colocación dependerá de la interfaz gráfica que se utilice. |
| GridViewRow | Sus dimensiones son dinámicas ya que depende la cantidad de información para asignarle una dimensión. Su colocación dependerá de la interfaz gráfica que se utilice. |
| Chart | Sus dimensiones son dinámicas ya que depende la cantidad de información para asignarle una dimensión. Su colocación dependerá de la interfaz gráfica que se utilice. |
| img | Sus dimensiones serán desde un ancho de 40px y una altura desde 40px dependiendo de la interfaz gráfica que se utilice. Su colocación dependerá de la interfaz gráfica que se utilice. |
| Login | Sus dimensiones serán desde 400px de ancho y 200px de altura dependiendo los componentes que se utilicen. Su colocación dependerá del tamaño de sus componentes y su interfaz gráfica. |

* 1. **Swing Containers**

|  |  |
| --- | --- |
| **Swing Containers** | |
| **Elemento** | **Definición del nombre** |
| Panel | No aplica. |
| HtmlForm | Sus dimensiones serán de acuerdo a los componentes que tiene dentro. Su colocación dependerá de la interfaz gráfica que se utilice. |
| header | Sus dimensiones serán de 100% de ancho y 100px de altura. Su colocación inicia desde la parte superior de la interfaz gráfica. |
| div | Sus dimensiones serán de acuerdo a los componentes que haya dentro. Su colocación dependerá de la interfaz gráfica que se utilice. |
| section | Sus dimensiones serán de acuerdo a los componentes que haya dentro. Su colocación dependerá de la interfaz gráfica que se utilice. |
| nav | Sus dimensiones serán de 900px de ancho y 80px de altura. Su colocación es 10px después de la parte superior de la interfaz gráfica y 300px de la izquierda. |

* 1. **Swing Menus**

|  |  |
| --- | --- |
| **Swing Menus** | |
| **Elemento** | **Definición del nombre** |
| **LoginStatus** | Sus dimensiones serán 52px de ancho y 49px de altura. Su colocación es 28px abajo de la parte superior de la interfaz gráfica y 833px de la izquierda. |

1. **Ubicación de Elementos**
   1. **Ubicación Gráfica**

**4.1.1 Captura de Mantenimiento a Bitácora**





1. **Reglas de Oro en el Diseño de Interfaces de Usuario**
   1. **Darle al Usuario el Control**

La interfaz debe permitir que trabajen dentro de un número limitado de tareas, así como también darles un cierto grado de control y flexibilidad para realizar sus tareas de manera rápida, confortable, y eficientemente.

* **Utilizar modos juiciosos**

Permitir que los usuarios elijan el modo particular de realizar las acciones en el sistema, en lugar de forzarlo en un solo modo.

* ***Permita que el usuario utilice el teclado o el ratón (flexible)***

Los usuarios tienen diferentes ánimos para utilizar el teclado o Mouse y pueden utilizarlo alternadamente, algunas funciones son más sencillas de usar con el Mouse pues reducen tiempo de ejecución.

* ***Permitir que los usuarios cambien de foco (interruptible)***

No forcé a los usuarios a terminar secuencias predefinidas. Deles las opciones a la cancelación o aceptación y permítale regresar a su opción anterior.

* ***Desplegar mensajes descriptivos (helpful)***

Utilizar términos que los usuarios puedan entender, en lugar de términos técnicos.

* ***Proporcionar inmediatas acciones reversibles y de retroalimentación (forgiving)***

El uso de los indicadores para retroalimentar el proceso es uno de aspectos sutiles de la de un interfaz.

* ***Indicar que es lo que puede hacer en el sistema (undo y redo)***
* ***Provide meaningful paths and exits (navigable)***

Permitir a los usuarios navegar a través de la interfaz. Proporcionarles caminos para llegar a cualquier lugar a donde desean (forward, back, etc.)

* ***Acomodar a los usuarios en diversos niveles de habilidad (accessible)***

Los usuarios de diferentes niveles deben poder interactuar con el programa en diferentes niveles. Configurar interfaces de acuerdo a su nivel de interacción con el sistema (barras estándar o avanzadas).

* ***Hacer la interfaz transparente (facilitative)***

Permitirles a los usuarios centrarse en su tarea utilizando objetos que lo centren en ellas, en lugar de tener que traducir lo que deben hacer.

* **Permitir a los usuarios configurar la interface (preferences)**

Permitirle al usuario configurar la presentación de su información (colores, letras, tipos de vista).

* **Invitar al usuario a interactuar con los componentes, la interfaz debe ser explorable.**

* 1. **Reducir la carga de memoria al usuario**
* ***Relevar la memoria de corto plazo (remember)***

No forzar a los usuarios a recordar y repetir lo que la computadora puede hacer por ellos (copy, paste). Manipular información que pueden utilizar en diferentes lugares.

* ***Acelerar el reconocimiento, no recordar (recognition)***

Que al ver el usuario la interfaz pueda recordar. Por ejemplo Ponerle una lista con datos previos (equipos, ciudades, tipo de cliente, etc.)

* ***Proporcionarle pistas visuales (inform)***

Proporcionarle información de dónde, que está haciendo y que puede hacer (titulo de la barra de la ventana, blink cursor, status bar, etc.)

* ***Proporcionarle defaults, undo, and redo (forgiving)***

Utilizar múltiples versiones, y poder sustituirlas (la posibilidad de guardar versiones y nombrarlas con diferentes nombres).

* ***Proporcionarle atajos (shortcuts) en la interface (frequency)***
* ***Llave de acceso Mnemonica (File, Edit) accelerator (short key) combinación de teclas.***
* ***Promover la acción-objeto (intuitive)***

Consiste en que le permita al usuario relacionar entre los objetos y las acciones del sistema.

* ***Utilizar metáforas del mundo real (transfer)***

Transferir el conocimiento sobre lo que ya conoce y aplicarlo en el trabajo (utilizar dispositivos como micrófono, teléfono, etc.)

* ***Utilizar el acceso progresivo (context)***

Facilitarle el acceso a las características comunes y acciones usadas frecuentemente. Mostrarle que es lo que necesita, cuando lo necesita y donde lo quiere. No se requiere mostrarle todas las funciones del producto, puede ser gradualmente.

* ***Promover la claridad visual (organize)***

Organizar, agrupando ítems, listas, enumerando, usando en encabezamientos, o cajas de texto.

* 1. **Hacer Consistente la Interface**
* ***Sostener el contexto de la tarea del usuario (continuity)***

Proporcionarles puntos de referencia cuando esté navegando por el sistema. Títulos en las ventanas, navegación en mapas o árboles, vista dinámica de donde se encuentra. Predecir los resultados de las acciones.

* ***Mantener la consistencia dentro y fuera de los productos (experience)***

Que los conceptos que ya conocen de otros sistemas los puedan aplicar en diferentes situaciones en diferentes programas o parte del programa. Como introducir nuevos comportamientos, cuando se amplía la funcionalidad de una interfaz, los usuarios tienen que aprender estos nuevos comportamientos que no deben ser muchos.

* ***Que se produzcan resultados iguales en la interacción (expectation)***

Si los resultados del diseño serán diferentes de lo que el usuario espera, informarle antes de que ejecute la acción.

* ***Proporcionar apariencia estética e integridad (attitude)***

Que el diseño sea presentado como si una sola persona lo hubiera realizado. Mantener la integridad de la información y consistencia colores letras iconos, etc.

* ***Animar a la exploración (predictability)***

Interfaces amigables, intuitivas, predecibles, etc.