**Hugo Daniel Rojas Fuentes (729 2 30 3522, unihugodaniel@gmail.com)**

**Alan Alexis Velázquez Romero(alanalexisvelazquezromero@gmail.com)S6**

**Componente: Interfaz Administrador**

**Objetivo**

Crear la interfaz “Administrador” para poder gestionar los datos de los médicos, todo esto con el contenido apropiado que se debe de contemplados, ya que, deben ser los más necesarios para que se tenga la garantía de realizar las siguientes acciones; dar de alta a un nuevo médico, modificar sus datos, dar de baja y consultar toda la información, se logara almacenar en una Base de Datos, que solo tenga la información del médico, para eficiencia y organización de todos aquellos datos que se logren ingresar a la página. Por lo cual se está haciendo el apartado del administrador, para conocer lo que entra a la página sabiendo que se está realizando.

**¿Con qué tecnología lo hacemos?**

Utilizaremos el editor de código NetBeans 8.2 (para la elaboración de interfaces, conexión con Base de Datos y diseño de documentos HTML con lenguaje CSS)se utilizara primamente para el diseño de las interfaces o en este caso de la interfaz de administrador, además de la programación apropiando para el funcionamiento de cada uno de los campos a llenar del médico, con la ayuda de un generador de Base de Datos GlassFish 5.0 y el gestor de Base de Datos XAMP última versión (7.1.33 [consultado por última vez el 22/11/2019 a las 18:24 hrs]) estos son plataformas que utilizaremos para almacenar toda la información de los médicos previamente solicitados.

**¿Cómo lo hacemos?**

Lo haremos con la ayuda del lenguaje HTML, aquí escribiremos una serie de instrucciones con el formato de código, mismo que sigue una estructura donde se asigna un título y un cuerpo de la interface, que es la vista de una página web, después por medio de la utilización del lenguaje CSS, personalizaremos los componentes de la interfaz a nuestro gusto o con los colores y estilos de la fuente solicitados por el encargado de diseño.

Con JavaServer Pages (JSP) la cual es una tecnología que ayuda a los desarrolladores de [software](https://es.wikipedia.org/wiki/Software) a crear [páginas web](https://es.wikipedia.org/wiki/P%C3%A1gina_web) [dinámicas](https://es.wikipedia.org/wiki/Contenido_din%C3%A1mico) basadas en [HTML](https://es.wikipedia.org/wiki/HTML) y [XML](https://es.wikipedia.org/wiki/XML), entre otros tipos de documentos. JSP es similar a [PHP](https://es.wikipedia.org/wiki/PHP), pero usa el [lenguaje de programación](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n) [Java](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)).

Para desplegar y correr JavaServer Pages, se requiere un [servidor](https://es.wikipedia.org/wiki/Servidor) web compatible con contenedores [Servet](https://es.wikipedia.org/wiki/Servlet) como [Apache Tomcat](https://es.wikipedia.org/wiki/Apache_Tomcat) o [Jetty](https://es.wikipedia.org/wiki/Jetty). Todo esto se complementa con los servlets los cuales son  [clase](https://es.wikipedia.org/wiki/Clase_(inform%C3%A1tica))s en el [lenguaje de programación Java](https://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java), utilizada para ampliar las capacidades de un servidor. Aunque los servlets pueden responder a cualquier tipo de solicitudes, estos son utilizados comúnmente para extender las aplicaciones alojadas por servidores web, de tal manera que pueden ser vistos como [applets de Java](https://es.wikipedia.org/wiki/Applet_Java) que se ejecutan en servidores en vez de navegadores web. Este tipo de servlets son la contraparte Java de otras tecnologías de contenido dinámico Web, como [PHP](https://es.wikipedia.org/wiki/PHP) y [ASP.NET](https://es.wikipedia.org/wiki/ASP.NET).

Una vez que se tenga todas las plataformas ya mencionadas debemos de conocer los campos para dar de altas, bajas, consultas y modificaciones; los elementos y diseño de la interfaz se modificara para mejor el entendimiento de cada apartado:

|  |  |  |
| --- | --- | --- |
| **Identificación De**  **Requerimientos** | RF1 | |
| **Nombre Del Requerimiento** | Altas de medico | |
| **Características** | * Se mostrara la información del médico: Nombre, Apellidos, Curp, RFC, Cedula profesional, ID persona, ID médico. * Se mostrara y contendrá los campos a llenar del médico. * Verificación de cedula (Botón). | |
| **Descripción** | El llenado de todos estos campos será necesario y obligatorios para que se logre registrar con éxito todos, por lo que si no se llena alguno en el caso del nombre completo, ya que, desde este hasta curp, RFC, se hará una comparación de información al momento de consultar la cedula profesional del médico.  Este desde la página que se esté obteniendo la cedula esto para que se valide la información proporcionada, los campos de id se darán por el administrador para mejorar el control del personal, que este por entrar o solicitar su alta. Esto se logra con el botón de verificar el cual nos indica que la información fue consultada y aparvada además de que validara los campos anteriores para el registro en la página.  Cada una de los campos estará restringió con los elementos solo será necesarios, pues si este se solicitan números y letras en el caso de la curp, solo se ingresara los caracteres que son de la curp, así con todos los demás campos que se están dando el este apartado.  Una vez que se llenó todos los campos se hará la alta con el botón de aceptar, este tiene que guardada la información que se llenó, estos se almacenaran en una base de datos con los atributos que se solicitaron y desde la misma página el administrador podrá visualizar la o las personas que dio de alta, esta con las columnas de cada campo ya mencionado. También habría un botón de cancelar, este eliminara y no permitirá el registro del médico en curso. Por último se dará un mensaje que el registro se logró con éxito, al momento de que este salga se limpiara los campos para un nuevo registro para almacenar. | |
| **Actores** | Desarrolladores, Medico, Administrador. | |
| **Prioridad de Requerimientos** | | **Exigible de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RFU2 |

Relación de casos a tomar en cuenta al momento de haces los requerimientos:

**Caso 1:**

Si una persona ya está registrada ya no se volverá a registrar de nuevo por eso se hace la comparación con la tabla de base de datos, si este existe no dejara que haga otro registro.

**Case 2:**

De no tener todos los datos correctos el registro no le permitirá ingresar lo que tenga ya que se está incumpliendo lo que se requiere.

**Case 3:**

En caso de que la cedula este en trámite se deberá de hablar con la persona adecuada pues es un requisito para saber si la persona que se está registrando está dando información correcta.

**Caso 4:**

Los botes no logren hacer su función de acuerdo a lo que fueron programados;

Botón guardar: no mande la información de los campos a la bases. Esto también podría ser por una mala conexión con la base de datos, por lo cual no dejara que entre los datos.

Botón de verificación: no se realiza la función de concatenar la información y dar un veredicto de que si es o no correcto lo que se está dando por la persona.

Botón cancelar: no elimina los datos ingresados.

Botón aceptar: al momento de que se despliegue el mensaje al momento de dar click debe de mandar la información.

|  |  |  |
| --- | --- | --- |
| **Identificación De**  **Requerimientos** | RF2 | |
| **Nombre Del Requerimiento** | Bajas de medico | |
| **Características** | * Se mostrara la información del médico que se va a eliminar: Nombre, Apellidos, Curp, RFC, Cedula profesional, ID persona, ID médico. * Llenado de los datos que se van a eliminar. * Eliminar registro (Botón). | |
| **Descripción** | En este apartado se solicitara que el administrador seleccione el id de la persona que se desea eliminar de los registro del médico, esto se mostrara en los campos antes aplicados en la “alta de médico”, esto se hará para visualizar a la persona que desea eliminar y no cometer errores, una vez que este seguro de la persona que es se ira al botón de eliminar este hará la función de preguntar ;si está seguro con un mensaje que le dará aceptar o cancelar si así lo desea, una vez hecho esto borra esta información se eliminara de forma permanente de los registro de las tablas de base de datos . | |
| **Actores** | Desarrolladores, Medico, Administrador. | |
| **Prioridad de Requerimientos** | | **Exigible de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RFU2 |

**Caso 1:**

El campo de id está configurado para validar si la persona existe o no, si esta existe mostrara la información de aquella persona que se borrara.

**Caso 2:**

Si no se dará un mensaje que se escribió mal el id, que vuelva intentar y si no selecciono ninguna columna, o que el id no existe, por lo tanto no mostrara nada.

**Caso 3:**

El botón no logre eliminar la información requerida.

**Caso 4:**

La tabla que muestra a los elementos que se registraron no muestre los que ya no existen.

|  |  |  |
| --- | --- | --- |
| **Identificación De**  **Requerimientos** | RF3 | |
| **Nombre Del Requerimiento** | Modificaciones de datos | |
| **Características** | * Se mostrara los campos que desea cambiar del médico: Nombre, Apellidos, Curp, RFC, Cedula profesional, ID persona, ID médico. * Llenado de los datos que se van a modificar de médico. * Modificaciones (Botón). | |
| **Descripción** | En este apartado con dar un click en la tabla, con la fila de la persona que desea modificar o actualizar la información que se dice cambiar esto mostrando los datos que ya se tiene en existencia que se tiene de la persona esto con la ayuda de los campos llenados anteriormente, para ello se debe estar en existencia en los registro previos (altas), una vez que se ingresó se dará la acción de modificar todos los elementos que se desean.  Los campos que no se tiene permitido será la célula, por los limitantes de campos. Se enviara un mensaje, el cual se dirá que si está seguro de actualizar la información; si acepta la información se guardara en el apartado de la persona que selecciono. | |
| **Actores** | Desarrolladores, Medico, Administrador. | |
| **Prioridad de Requerimientos** | | **Exigible de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RFU2 |

**Caso 1:**

Si se ingresó un id incorrecto no permitirá ver nada de los elementos que se tienen, no dará acceso a la información.

**Caso 2:**

No se selección la fila deseada, no se mostrara los campos que se desean.

**Caso 3:**

Validación de campos que no permite cambiar los datos incorrectos (cedula, RFC, Curp).

|  |  |  |
| --- | --- | --- |
| **Identificación De**  **Requerimientos** | RF4 | |
| **Nombre Del Requerimiento** | Consultas de médicos (datos). | |
| **Características** | * Mostrará la tabla de todos los médicos ya registrados con los campos llenados. * Búsqueda (Botón). | |
| **Descripción** | En la parte superior de la tabla a que contiene todos los registros, dará la opción de la búsqueda por medio de id, nombre o por medio de curp.  Una vez seleccionada esta opción se escribirá el registro que se desea saber la información, con el botón de búsqueda hará la comparación de lo escrito con lo que esta en las tabla de registro, se mostrara su nombre, apellido Paterno, apellido Materno, curp, rfc, cedula profesional, id persona, id medico (pueden ser todas, unas cuantas, o una). | |
| **Actores** | Desarrolladores, Medico, Administrador. | |
| **Prioridad de Requerimientos** | | **Exigible de Requerimientos** |
| Alta | | Exigible |
| **Requerimientos relacionados** | | RFU2 |

**Caso 1:**

Cuando se busca por ID de persona, solo mostrara un solo dato ya que no hay 2 o más con ese mis código de números.

**Caso 2:**

La búsqueda por nombre dará todas las personas que tengan ese nombre.