**Proyecto Programación IV valor 35%**

**Descripción general**

Un centro costarricense de investigación, en compañía con una empresa desarrolladora de *software*, han propuesto el desarrollo de una aplicación *web* que sirva para ubicar mediante un mapa digital los últimos eventos sísmicos importantes, y que también guardar la información para futuras consultas. El sistema mostrará los eventos sísmicos globales.

Todas las páginas tendrán el mismo estilo, para lo que se creará una o varias hojas de estilo CSS, de manera que se mantenga consistente en todo el sitio *web*: en todas las páginas debe aparecer en la parte superior el logo y el nombre del centro de investigación; a la izquierda del sitio aparecerá una barra de navegación; a la derecha saldrán noticias recientes (que serán cargadas desde la base de datos) y al final de la página aparecerán los créditos de la empresa desarrolladora.

El personal de la empresa considera que es necesario definir las siguientes vistas:

**Página de inicio.** En la que se incluya información básica del sitio. La página de inicio mostrará opciones para que los usuarios puedan ingresar al sitio o registrarse como usuarios nuevos.

**Formulario de registro.** Por motivos de control y para poder tener información estadística de uso del sitio, solamente aquellas personas que estén registradas en el sistema podrán ver la información completa del sitio. Es por ello que habrá un formulario donde se registre el nombre, apellidos, dirección de correo, teléfono, contraseña y fecha de nacimiento del visitante. El sistema validará cada uno de los campos del formulario antes de efectuar el envío de los datos y volverá a realizar la verificación en el servidor antes de registrar la información. Todos los datos se guardarán en una base de datos en el mismo servidor.

Para verificar la contraseña se escribirá dos veces en campos separados de tipo *password*, mientras se escribe la contraseña el sistema debe indicar la fortaleza de la misma. Posteriormente se le mostrará una página al usuario indicado que la información se registró correctamente. Para ello, NO deberá mostrar ninguna caja de diálogo. No existirá tampoco ningún proceso de validación posterior de la cuenta.

El formulario de registro y actualización de datos del usuario permitirá cargar una fotografía del usuario, la cual será almacenada en un directorio específico en el servidor. La foto no debe guardarse en la base de datos, pero sí la ruta en el servidor, de manera que pueda recuperarse para mostrarla en la página. El nombre del directorio y las convenciones para el nombre del archivo quedan a criterio de los estudiantes. Tengan en cuenta que varios usuarios pueden subir archivo con el mismo nombre, por lo que deberán considerar algún mecanismo para poder distinguir los archivos.

**Formulario de ingreso (*login*).** Se verificará la existencia del usuario y se comprobará su clave. El sistema debe almacenar la última fecha y hora de cualquier usuario en el momento de ingresar al sistema. Esta información la podrá consultar el administrador del sistema.

**Página de administración.** Existen dos tipos de usuario: administradores y visitantes. Los administradores pueden ver y eliminar usuarios visitantes. Además, los administradores podrán incluir, modificar y eliminar las noticias recientes, las cuales serán visualizadas en todas las páginas del sistema, mostrando primero las noticias más recientes (es decir, de fecha más próxima). Los usuarios administradores están fijos en el sistema y no pueden modificarse.

**Página para modificar datos personales.** Un usuario (tanto administrador o visitante) puede modificar su nombre, fecha de nacimiento y contraseña, usando las mismas reglas de validación que en el formulario de registro.

**Página del mapa.** Implementar una funcionalidad que permita geolocalizar los epicentros en un mapa interactivo. Pueden hacer uso de un api por medio del cual se mostrará la información sísmica obtenida como por ejemplo USGS Earthquake API(<https://earthquake.usgs.gov/fdsnws/event/1/>). Los datos a recopilar deben incluir, como mínimo, la latitud, longitud, magnitud, profundidad, fecha y hora del evento sísmico. Pueden desarrollar una interfaz web interactiva utilizando un framework adecuado (por ejemplo, Flask o Django). La interfaz debe mostrar un mapa en tiempo real con marcadores en los epicentros de los eventos sísmicos. Deberán implementar filtros de búsqueda para que los usuarios puedan consultar datos por magnitud, fecha, y ubicación.

El mapa debe actualizarse en tiempo real con los nuevos datos obtenidos de la API sísmica. Pueden utilizar técnicas de polling o websockets para actualizar los datos y la visualización sin necesidad de recargar la página.

**Documentación y Entrega**

1. Documentar todo el proceso de desarrollo, incluyendo instrucciones de instalación, configuración y uso de la aplicación.
2. Entregar el código fuente completo junto con la documentación y una presentación que explique el funcionamiento y las características de la aplicación.

**Criterios de Evaluación**

1. Precisión y Eficiencia: La aplicación debe funcionar sin errores, proporcionando datos precisos y actualizaciones en tiempo real.
2. Interfaz de Usuario: La interfaz debe ser intuitiva, fácil de usar y visualmente atractiva.
3. Calidad del Código: El código debe ser limpio, bien comentado y seguir las mejores prácticas de desarrollo.
4. Documentación: La documentación debe ser clara, detallada y facilitar la comprensión y uso de la aplicación.
5. Innovación: Se valorarán características adicionales que mejoren la funcionalidad y usabilidad de la aplicación.

**Herramientas a utilizar**

1. Python: Para el desarrollo backend y manejo de datos.
2. Flask/Django: Para la creación de la aplicación web.
3. Google Maps API / Leaflet.js: Para la visualización de mapas.
4. SQL Server: Para la gestión de la base de datos.
5. USGS Earthquake API: Para la obtención de datos sísmicos.