

Ejercicio SISU Technologies

Planteamiento del problema

Uno de los casos de uso de una determinada aplicación web es pedir que los usuarios dejen la siguiente información:

- Nombre
- Edad
- Ciudad

Esta información será capturada dentro de una forma.

Actualmente se cuenta con una serie de servicios web desarrollados que realizan las siguientes funciones:

- Obtener el listado de países disponibles.
- Obtener el listado de estados de un país (por medio del identificador del país).
- Obtener el listado de ciudades de un estado (por medio del identificador del estado).
- Guardar los datos del usuario.

Tarea principal

Desarrollar una página usando únicamente HTML, Javascript y CSS que cumpla con la interacción anterior. La solución deberá de tener el comportamiento mostrado en el video adjunto **como mínimo**.

Para el ingreso del país, el estado y la ciudad se utilizarán controles de dropdown. La interacción será de la siguiente forma:

- Inicialmente los dropdowns de estado y ciudad deberán de estar vacíos o no existir.
- El dropdown de país deberá de inicialmente estar poblado con los países.
- Al seleccionar un ítem del dropdown de países se debe de popular el dropdown de estados con los estados del país seleccionado.
- Al seleccionar un ítem del dropdown de estados se debe de popular el dropdown de ciudades con las ciudades del estado seleccionado.
- Se debe de validar:
 - El nombre debe de ser de 50 caracteres máximo y únicamente pueden ingresar letras (considerar acentos), espacios y puntos.
 - La edad debe ser entre 18 y 99 años (inclusivo).
- Al someter la forma se deben de enviar los siguientes datos al servicio web para guardar:
 - Nombre
 - Edad
 - Identificador de la ciudad

Una vez sometida la información la página debe mostrar el mensaje regresado por el servicio web.

Consideraciones técnicas

- La solución debe de utilizar comunicación asíncrona entre la página y el servidor (AJAX) para poblar todos los dropdowns y hacer el envío de los datos del usuario.
- Se deben de validar los datos ingresados en la página antes de enviarlos a los servicios web.
- No hay restricciones de marcos de trabajo o librerías posibles a utilizar, siempre y cuando sean de Javascript y sin necesidad de instalar algo adicional.
- El header de Content-Type de toda la comunicación hecha por POST debe ser "application/json", de lo contrario no responderá el backend.

Aspectos a evaluar

Los siguiente son puntos que serán evaluados en la solución:

- Cumplimiento de todos los requerimientos.
- Que no haya errores.
- Atención al detalle.
- Claridad del código.
- Documentación de código.
- Diseño lógico y gráfico.
- Creatividad y calidad de la solución.
- El tiempo que tomó realizar la prueba (esto se reporta al entregarla).
- Decisión de hacer la tarea adicional y su calidad (ver más abajo).

Tarea adicional

Esta tarea no es obligatoria, pero será tomada en consideración si se envía junto con la solución.

- Modificar el backend para que los datos de los nuevos usuarios que se reciban sean guardados en una base de datos. Puede ser MySQL, PostgreSQL o una base de datos en memoria como H2.
- Entregar un ejecutable compilado con estas modificaciones y su respectivo código fuente.

Material a entregar

- Archivos HTML, CSS y Javascript con la solución.
- Todos los archivos que se entreguen deben estar listos para usarse/montarse/correrse. Para evaluarlo, no deben requerir realizar algún build.
- Si para evaluar se requiere hacer algo además de abrir el archivo index.html, favor de adjuntar un readme con explicaciones.
- Entregarlos usando el dashboard de candidato desde donde se descargaron.
 - Importante especificar cuántas horas se invirtieron en el ejercicio. Se espera honestidad y transparencia.
 - Importante enviar esto antes del deadline que establece la plataforma, de lo contrario no se continuará en el proceso.
- En caso de **no** hacer la tarea adicional, por favor no incluyan el archivo ejecutable .jar y respectivo código fuente del backend.
- En caso de **sí** haber hecho la tarea adicional, incluir además del código fuente, la versión **compilada** del archivo .jar, lista para correrse.

Instalación de los servicios web (backend)

Los servicios web están desarrollados usando Spring. Se hace uso de Spring Boot para la ejecución, por lo que no es necesario instalar un servidor de aplicaciones.

Requerimientos

- JRE (Java Runtime Environment) 8 para ejecutar el backend
- JDK (Java Development Kit) 8 y Maven para compilar el backend

Instrucciones

Existen 2 carpetas:

- /fuente_servidor: Contiene el código fuente del backend
- /ejecutable: Contiene el backend en un archivo ejecutable

Ejecución

Para correr el servidor con el backend simplemente hay que ejecutar el archivo .jar en la carpeta de /ejecutable, usando:

```
java -jar challengeajax-0.0.1-SNAPSHOT.jar
```

Esto levantará el servidor de forma local en el puerto 8080. No debe de existir algún otro servicio corriendo en ese puerto.

```
http://localhost:8080
```

Edición

El código fuente del backend se encuentra en la carpeta /fuente_servidor. En caso de modificarlo se puede compilar y correr el servidor en modo de desarrollo usando:

```
mvn spring-boot:run
```

Como nota adicional, si se desea re-empacar el código como un archivo .jar como el que está en la carpeta /ejecutable, hay que usar el comando:

```
mvn clean package
```

Esto creará una carpeta /target con el archivo jar compilado.

Documentación y URLs de los servicios web

Para visualizar la documentación de los servicios web hay que ir a la dirección:

```
http://localhost:8080/docapi/index.html
```

Es muy importante revisar la documentación de los servicios web para hacer el ejercicio.

Se puede encontrar en esa dirección la funcionalidad de cada servicio web, la URL y el método (GET o POST) aceptado por cada uno. Adicionalmente se pueden hacer pruebas con ellos usando diferentes parámetros.

El nombre del API es "servicio".