



ELECTIVA DESARROLLO DE APLICACIONES WEB

MICROPROYECTO 1: FrontEnd

1. Objetivos:

- Reforzar los conceptos más importantes para la implementación de aplicaciones web desde el Front End basadas en cualquier Framework tal como Angular, React o Vue.

2. Prerrequisitos:

Los conocimientos a evaluar son:

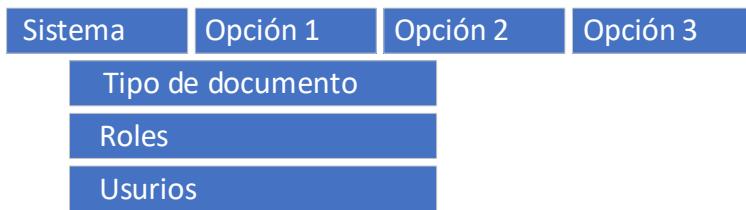
- Uso de Angular, React o Vue.
- Manejo de base de datos
- Validación de datos en el frontend.
- Seguridad.

3. Descripción:

- Se quiere desarrollar una aplicación web de tipo Single-Page Application (SPA) que permita administrar el módulo de usuarios del sistema con las siguientes características:
 - Para ingresar se debe usar login y password. El login será una cadena de texto con una longitud máxima de 40 caracteres. Para la validación, no se distingue entre mayúsculas y minúsculas. El password es una cadena con una longitud máxima de 200 caracteres, porque se debe guardar cifrado utilizando el algoritmo RSA con un cifrado mínimo (secret key) de 256bits.
 - La información mínima del usuario es: Nombres, Apellidos, tipo de documento de identidad, numero de documento de identidad, género, dirección de correo electrónico, número de teléfono móvil, rol dentro de la aplicación, fecha de nacimiento y foto.
 - Después de ingresar a la aplicación, se debe mostrar una página sencilla de bienvenida. Desde ahí debe tener acceso al menú de la aplicación en cascada de tipo horizontal como el que se muestra en la siguiente figura, a manera de ejemplo.



Menu en cascada



- El alcance del proyecto debe abarcar las opciones de la figura anterior: Gestión de los tipos de documento (Tipo de documento), Gestión de los roles (Roles) y Gestión de usuarios (Usuarios).
- El proceso de gestión significa que se debe poder crear, modificar, eliminar o consultar cualquier elemento, por ejemplo, un rol, un usuario o un tipo de documento.
- Los tipos de documento solo manejaran dos campos, el id o código, y el nombre o descripción del tipo de documento. Adicional a ello, y de manera automática, se deben registrar: la fecha y hora del evento (solo creación, modificación o eliminación) del elemento, y el usuario que realizo la acción.
- El proceso de eliminación no será equivalente al borrado físico de la información, sino a la deshabilitación o inactivación del registro en la base de datos, a través de un campo booleano llamado estado. Las demás operaciones (crear, modificar, eliminar) deben actuar de acuerdo con ese estado.
- Si se dese crear un nuevo elemento que coincide con un elemento inactivo, la aplicación debe advertirle al usuario, que el dato ya existe y preguntar si desea reactivarlo. La modificación o eliminación solo será posible sobre elementos activos. Solo se pueden visualizar los elementos activos. Se debe tener una interfaz de usuario aparte para ver y reactivar los elementos inactivos (eliminados):
 - Controlar el acceso a las diferentes interfaces de la aplicación con seguridad. Ningún usuario puede acceder directamente a cualquier interfaz directamente con la URL, sin estar previamente autenticado.
 - Los formularios de datos deben validar los campos de infromacion de acuerdo con la definición de los campos en la base de datos. Así, por ejemplo, se deben tener



en cuenta:

- Campos obligatorios
- Campos solo numéricos
- Validar formato de correos electrónicos
- Caracteres especiales
- Rango y formato de fechas
- Longitud de los campos de datos
- Rangos de valores permitidos
- Etc.
- Debe existir un rol inicial de Super Administrador que tiene acceso a todo. Y solo debe permitir dos usuarios, máximo, con ese rol. No debe permitir la asignación de ese rol a más usuarios. Debe advertir de esa situación a través de mensajes emergentes.
- Los nuevos roles no deben tener acceso al módulo de gestión de usuarios, por defecto.
- Para crear un usuario, deben existir tipos de documentos y roles ya registrados en el sistema.
- La implementación debe ser de tipo SPA (Single-Page Application). Las diferentes interfaces auxiliares se deben crear a partir de ventanas modales.
- Fecha entrega: por definir. Usar un repositorio de código como GitHub
- Enviar la URL con permiso de acceso.

Nota importante: verificar los recursos y condiciones necesarios para ejecutar la aplicación a partir de la clonación del repositorio.

Javier Alexander Hurtado

Departamento de Telemática

Universidad del Cauca