

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Universidad Tecnológica Nacional Facultad Regional Avellaneda** | | | | | | | | | | | | |
| Técnico Superior en Programación - Técnico Superior en Sistemas Informáticos | | | | | | | | | | | | |
| Materia: Laboratorio III | | | | | | | | | | | | |
| Apellido: |  | | | | Fecha: | | | | 01/07/2024 | | | |
| Nombre: |  | | | | Docente(2): | | | | Baez/Bongioanni | | | |
| División: |  | | | | Nota(2): | | | |  | | | |
| Legajo: |  | | | | Firma(2): | | | |  | | | |
| Instancia(1): | **PP** |  | **RPP** |  | | **SP** | **X** | **RSP** | |  | **FIN** |  |

1. Las instancias válidas son: 1er Parcial (**PP**), Recuperatorio 1er Parcial (**RPP**), 2do Parcial (**SP**), Recuperatorio 2do Parcial (**RSP**), Final (**FIN**) . Marque con una cruz.
2. Campos a ser completados por el docente.

## Realizar una ABM de Cryptos.

* + Desarrollar una aplicación que cuente con una única página donde exista el listado de las monedas dadas de alta (mostrar un spinner mientras cargan), así como un formulario para dar de alta a nuevas monedas. Para esto deberá usar HTML y CSS para darle agradable “look and feel”.
  + La lista de monedas deberá ser generada dinámicamente (con código JS) en base a la estructura del objeto utilizado, además deberá contar con algún manejador de eventos, tal que al hacer click en algún elemento de la lista, se carguen los valores la moneda seleccionada en el formulario.

En su defecto, contar con columna extra y que dentro tenga los botones necesarios para permitirnos la edición o eliminar la moneda

Se deberán habilitar los botones necesarios para permitirnos confirmar edición, cancelar edición o eliminar la moneda.



## Especificaciones header

1)Agregar un Navbar 2)Agregar Nombre del sistema

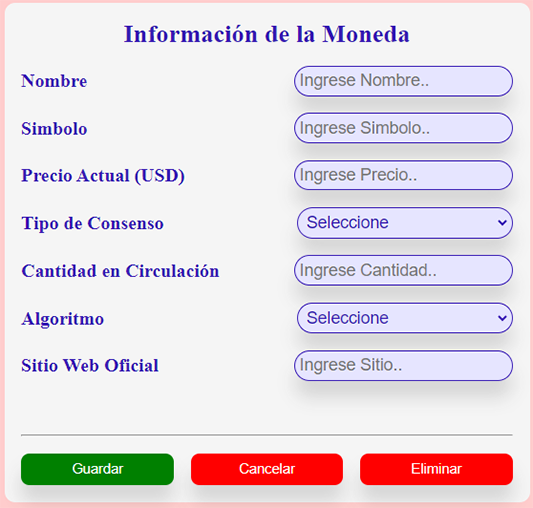
## titulo

******

1)Agregar un encabezado

2)Una foto relacionada al Sistema.

***form***

******

1)Respetar los controles de la imagen(el posicionamiento es a gusto). 2)Validaciones a todos los controles, en caso de error informar mediante alert

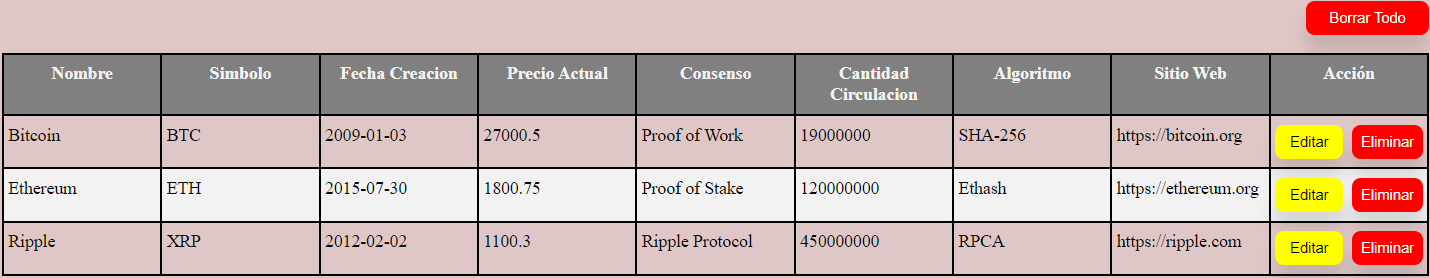
3)Las opciones de los select serán:

**Tipo de Consenso**: ‘Proof of Work’ y ‘Proof of Stake’

**Algoritmo**: ‘SHA-256’ , ‘Ethash’ , ‘Scrypt’ y ‘X11’

4)Mostrar el botón “Eliminar” solo si lo está Editando

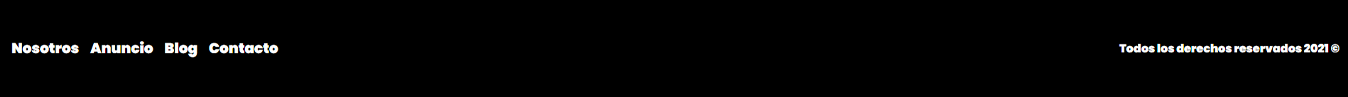
## tabla

******

1)Pintar el encabezado 2)Centrar tabla en la pantalla

3)Botón eliminar todos, utilizar confirm antes de ejecutar

## footer



1)Respetar posicionamiento

## spinner



1)Agregar spinner con icono del sistema

2)Agregar animación

# Javascript

1. Utilizar una clase “CryptoBase” que contenga los atributos:
   * id
   * nombre
   * símbolo
   * fechaCreacion
   * precioActual
2. Utilizar una clase “Crypto” que herede de “CryptoBase” con el resto de los atributos necesarios

# Backend

* + El backend será en localStorage. Agregar delay de 2.5 segundos en cada operación, de manera de poder visualizar el spinner.

**SEGUNDO PARCIAL**

**Map/Reduce/Filter**

* Agregar una sección que permita filtrar la tabla que se muestra por pantalla por tipo de algoritmo, y muestre por pantalla el promedio de precio solo del algoritmo seleccionado. Mostrar “N/A” si no hay filtros aplicados.



**Bootstrap**

* Utilizar mínimo de 10 clases de Bootstrap.

**Backend**

* Reemplazar la capa de acceso a datos por llamadas AJAX al servidor
* Utilizar xmlhttprequest en al menos dos de las cinco operaciones.
* El spinner debe ser visible mientras se realizan dichas operaciones.