TI-CAPITAL HUMANO

TEMA: MVC
SUBTEMA: Web API

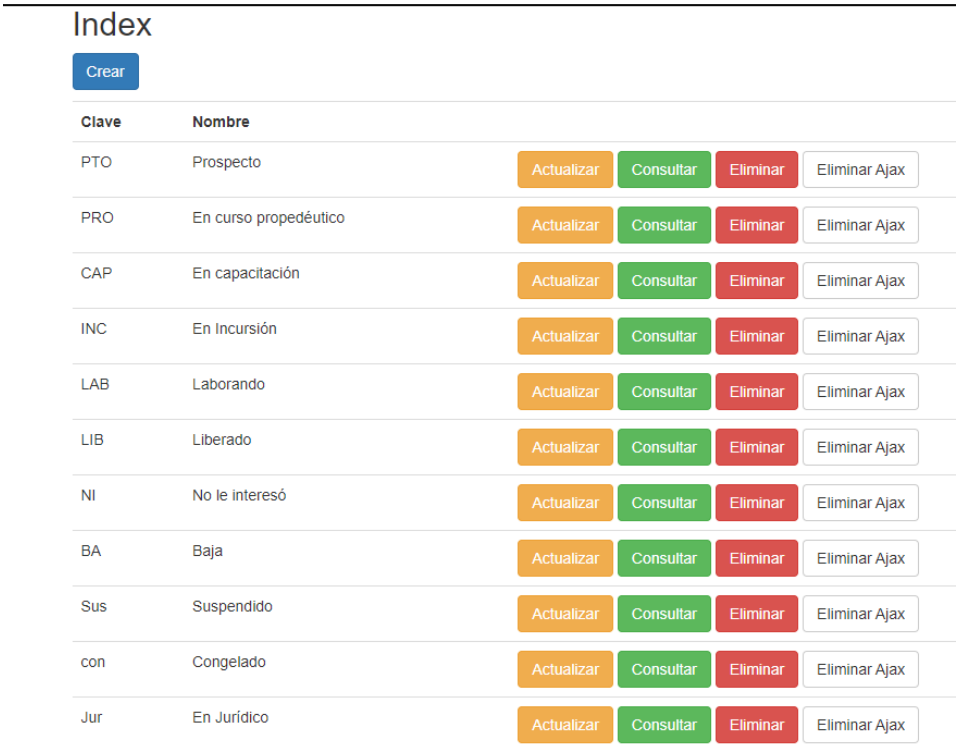
EJERCICIO – Crear una Web API y Consumirla desde el Servidor y desde el Cliente

OBJETIVO

Crear y Consumir una Web API para la actualización de la tabla EstatusAlumnos, desde el Servidor utilizando httpClient y desde el Cliente utilizando AJAX

DESARROLLO

Crear una Web API con los métodos CRUD para la actualización de la tabla EstatusAlumnos. Asimismo, agregar a la aplicación MVC el controlador y vistas correspondientes para que el usuario pueda realizar las actualizaciones a dicha tabla, quedando la siguiente interfaz.



1 Crear Web API.

1.1 Crear Aplicación tipo Servicio ASP .Net Core Web API.

Crear una nueva aplicación de tipo “ASP .Net Core Web API” con el nombre de **WebAPIEstatusAlumnos,** para el FrameWork .Net 6.0

1.2 Agregar EntityFramework

Agregar EntityFrameWork para la tabla EstatusAlumnos

1.2.1 Instalar los siguientes paquetes NuGet

Microsoft.EntityFrameworkCore

Microsoft.EntityFrameworkCore.Design

Microsoft.EntityFrameworkCore.SqlServer

Microsoft.EntityFrameworkCore.Tools

1.2.2 Ejecutar en la consola del administrador de paquetes el comando Scaffold-DbContext

Ejecutar en la consola del administrador de paquetes el comando Scaffold-DbContext, con la conexión y proveedor correspondiente a su base de datos, únicamente para la tabla EstatusAlumnos, la clase de contexto se debe llamar EstatusContext y debe quedar en la carpeta Models/Context, asimismo las entidades deben de quedar en la carpeta Models/Entidades

1.2.3 Configurar la cadena de conexión en el archivo appsetting.json

1.2.4 Configurar servicio del DBContext en el archivo Program.cs

1.3 Agregar el Controlador

Agregar el controlador de API con acciones que usan EntityFrameWork llamado EstatusAlumnosController

2 Probar el Web API

Probar cada uno de los métodos del WebAPIEstatusAlumnos a través del PostMan recordando que para ello deberán usar los verbos GET, PUT, POST, DELETE

3 Consumir WebAPI

3.1 Consumir el WebAPI desde el Servidor con HttpClient

3.1.1 En el proyecto de Negocio agregar la clase NEstatusAlumnos

Crear la clase NEstatusAlumnos con los métodos de acción CRUD para la tabla de EstatusAlumnos

a) List<EstatusAlumnos> Consultar()
b) Void EstatusAlumnos Consultar(int id)
c) EstatusAlumnos Agregar(EstatusAlumnos)
d) Void Actualizar(EstatusAlumnos)
e) Void Eliminar(int id)

Implementandolos accesando el Web API WebAPIEstatusAlumnos

3.1.2 En el proyecto de Presentacion agregar Controlador EstatusAlumnos

Crear en la carpeta Controllers del Proyecto de Presentación, el controlador EstatusAlumnosController con la plantilla Controlador de MVC 5 con acciones de lectura y escritura

Implementar cada uno de los métodos utilizando la clase NEstatusAlumnos

3.1.3 Crear las vistas correspondientes a cada uno de los métodos de acción agregados al Controlador EstatusAlumnos

Crear las vistas correspondientes a cada uno de los métodos de acción agregados al Controlador EstatusAlumnos, dando la apariencia de la pantalla indicada, sin el botón “Eliminar Ajax”, la cual se agregará posteriormente.

3.2 Consumir el WebAPI desde el Cliente con AJAX

3.2.1 Agregar el botón Eliminar AJAX

En la vista Index, agregar el botón **Eliminar AJAX,** mismo que deberá responder desde el cliente a fin de que mediante AJAX consulte el Estatus seleccionado, a través del WebAPIEstatusAlumnos y muestre la información en una ventana modal. La ventana modal deberá tener un botón para confirmar la eliminación del registro mostrado, y en su caso, proceder a eliminarlo mediante el llamado a la WebAPIEstatusAlumnos.

3.2.2 Agregar la ventana Modal para eliminar

Agregar la ventana Modal para eliminar, como la que se muestra en la figura

3.2.3 Agregar la función javascript para Consultar

Agregar la función javascript que se ejecutar cuando se dé click al botón “Eliminar AJAX”, misma que deberá llamar al WebAPIEstatusAlumnos por medio de AJAX, para consultar el elemento seleccionado y en la respuesta obtenida llenar la ventana modal y mostrarla.

3.2.4 Agregar la función javascript para Eliminar

Agregar la función javascript que se ejecutará cuando se dé click al botón “Eliminar”, de la Ventana Modal, misma que deberá llamar al WebAPIEstatusAlumnos por medio de AJAX, para Eliminar el elemento mostrado.

3.2.5 Adaptar el Web API WebAPIEstatusAlumnos para permitir Intercambio de recursos de origen cruzado (CORS)

Adaptar el Web API WebAPIEstatusAlumnos para permitir Intercambio de recursos de origen cruzado, para ello:

- Agregar los servicios y uso de las CORS en el **Program.cs**, indicando la url desde la cual se le permitirá hacerle llamadas.