

Documentación Prueba Técnica Datasoft

David González Villanueva

La estructura del proyecto se divide en dos partes complementarias, el proyecto API y el proyecto WEB.

Para la autenticación se usó la default de Identity que nos ofrece .net, esto nos permite tener una sección de **Registro** y **Login**.

Para el Registro tendremos la opción de insertar el nombre del usuario, el email, la contraseña y la confirmación de la contraseña.

Register

Create a new account.

Register

Para el Login solo ocuparemos el email y la contraseña, la opción de recordar usuario y la opción de olvidar la contraseña para poder recuperar la cuenta.

Log in

Use a local account to log in.

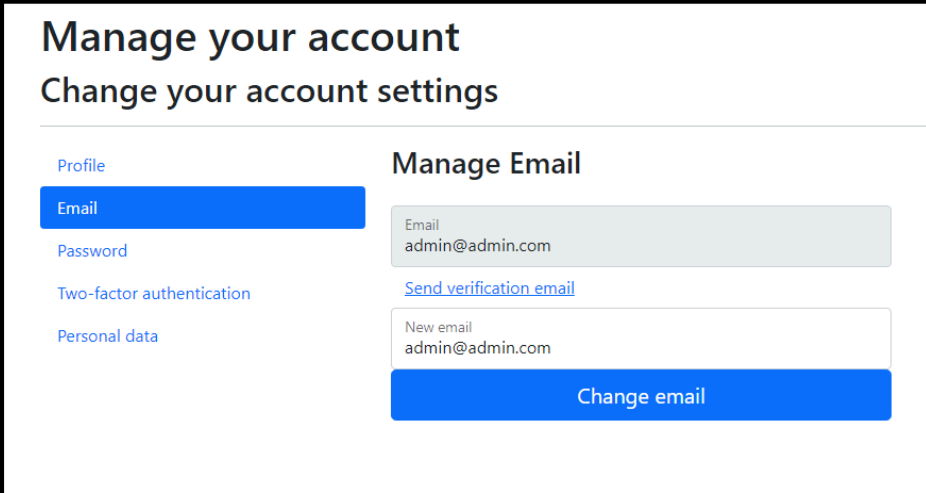
☐ Remember me?

Log in

[Forgot your password?](#)

[Register as a new user](#)

También de aquí se crea el perfil el cual se puede modificar la información del registro.



The screenshot displays a web interface for managing an account. The main heading is "Manage your account", followed by the sub-heading "Change your account settings". On the left, a vertical sidebar lists navigation options: "Profile", "Email" (highlighted in blue), "Password", "Two-factor authentication", and "Personal data". The main content area is titled "Manage Email". It contains a form with two input fields: the first is labeled "Email" and contains the text "admin@admin.com"; the second is labeled "New email" and also contains "admin@admin.com". A blue button labeled "Change email" is positioned below the second input field. A link labeled "Send verification email" is located between the two input fields.

Toda esta parte se encuentra en

Al registrarse se le asignará por defecto el rol de “Usuario”, como usuario las acciones que puede realizar son las de crear citas, editarlas, ver los detalles y eliminar citas.

Antes de crear una cita, se deben crear servicios, los cuales pueden crear empleados o admin. A la hora de crear un servicio, se podrá ingresar el nombre del Servicio, una descripción y el precio del servicio.

Create

Servicio

Nombre

Servicio 6

Descripción

Descripcion o detalles del servicio

Precio

1000

Create

[Back to List](#)

Los Servicios también se pueden editar, el cual la vista es similar al anterior y se podrá ver sus detalles y tambien estara la opcion de eliminar un servicio, siempre y cuando no haya una cita con el servicio en el momento, Al entrar al menú de citas se van a observar las citas que tenga en orden de cuál está más cercana, el estado de la cita, la fecha y el servicio que se solicitó, junto con las demás opciones.

Index

[Create New](#)

Estado	Fecha	Servicio	
Programada	10/1/2024 05:38:00	Servicio2	Edit Details Delete
Programada	3/2/2024 07:06:00	Servicio1	Edit Details Delete
Programada	26/2/2024 12:00:00	Servicio3	Edit Details Delete

Al entrar a crear se verá de la siguiente forma, se podrá seleccionar la fecha (día y hora) de la cita y el servicio que desea.

Create

Cita

Fecha

2024-02-15T19:06

Servicio

Servicio3

Create

[Back to List](#)

El edit al igual que es create, podrá editar la fecha de la cita y/o el servicio que se solicitó.

Edit

Cita

Fecha

2024-01-10T17:38

Servicio

Servicio1

Save

[Back to List](#)

Y para la vista del empleado, se puede editar el estado de la cita.

Edit

Cita

Estado

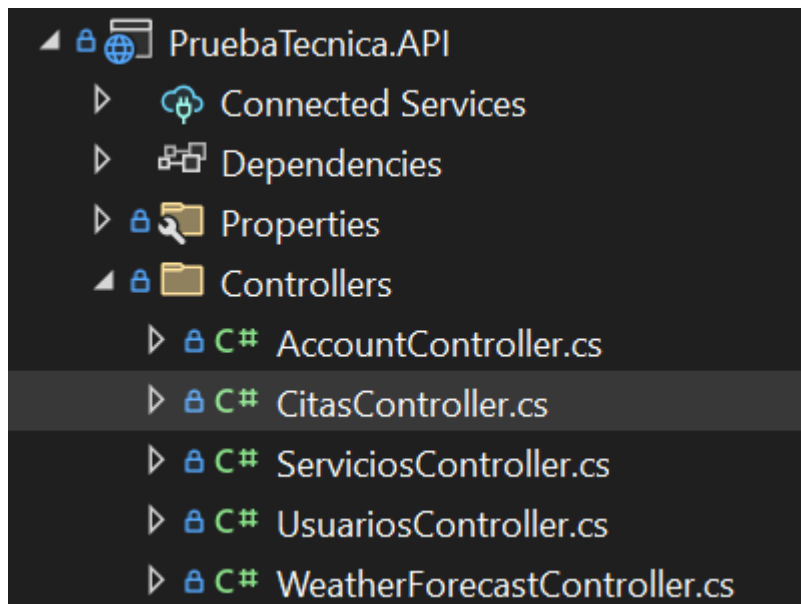
Programada

Save

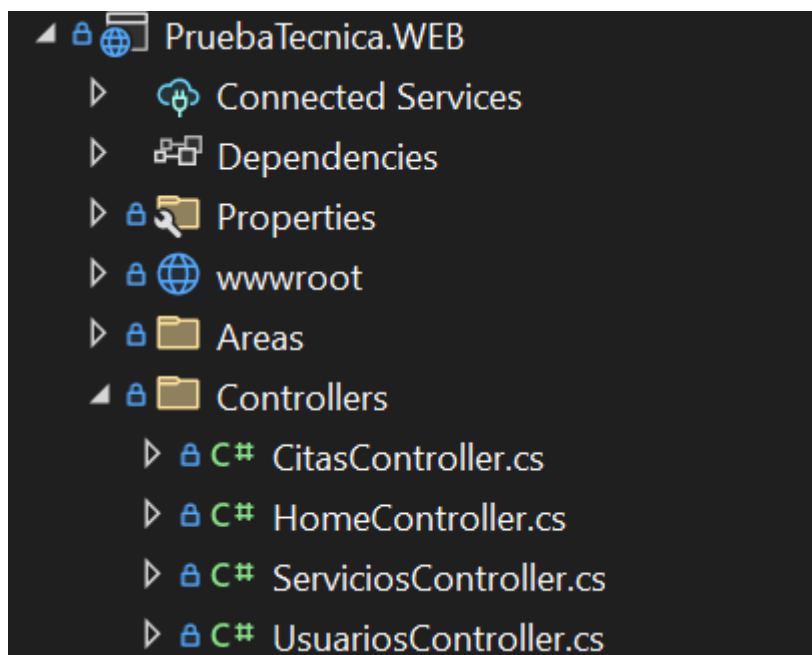
[Back to List](#)

Estas funciones son las mismas para los Usuarios, cambiando que la pestaña editar de los Usuarios es para poder editar el rol.

Estas funciones se hacen con controladores, en la parte del proyecto API se hacen las funciones de Put, Post, Get y Delete.



Y en la parte del proyecto WEB se especifican las funciones Edit, Details, Crear y Delete de las Citas, Usuarios, Servicios, etc.



Aunque las funciones Post, Put, Get y Delete se realizan con Ajax en las vistas respectivas de cada una. Con el siguiente código, el cual también despliega un mensaje en pantalla si la acción se realiza con éxito o se ocurrió algún error.

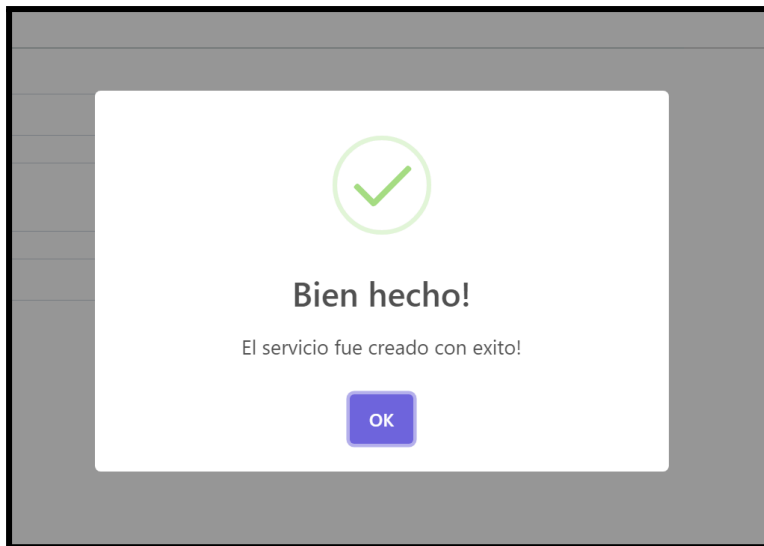
```

@section Scripts {
    <script>
        $("#formDelete").submit(function (e) {
            e.preventDefault();
            if ($('#formDelete').valid()) {
                $.ajax({
                    url: '/Citas/Delete/' + $("#CitaId").val(),
                    type: 'post',
                    data: new FormData(document.getElementById("formDelete")),
                    processData: false,
                    contentType: false
                }).done(function (success) {
                    if (success) {
                        Swal.fire({
                            title: "Bien hecho!",
                            text: "La cita fue borrada con exito!",
                            icon: "success"
                        }).then((result) => {
                            if (result.isConfirmed) {
                                window.location.href = "/Citas/Index";
                            }
                        });
                    } else {
                        Swal.fire({
                            title: "Error!",
                            text: "Algo salio mal!",
                            icon: "error"
                        });
                    }
                }).fail(function () {
                    Swal.fire({
                        title: "Error!",
                        text: "Algo salio mal!",
                        icon: "error"
                    });
                });
            }
        });
    </script>
    @{
        await Html.RenderPartialAsync("_ValidationScriptsPartial");
    }
}

```

Esta sección de código se encuentra en cada una de las vistas de las funciones que usen Put, Post o Delete de los Usuarios, Servicios y Citas.

Si la acción se realiza con éxito aparecerá el siguiente mensaje en pantalla.



Para la base de datos se usó el siguiente diagrama:

