UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES
Centro de e-Learning
Secretaría de Cultura y Extensión Universitaria

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Diplomatura en programación web full stack con React JS

UTN.BA UNIVERSIDAD TECNOLÓGICA NACIONAL FACULTAD REGIONAL BUENOS AIRES | Centro de e-Learning Secretaría de Cultura y Extensión Universitaria

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Módulo 6:

Administración, API Rest y despliegue

Unidad 2:

CRUD (parte 2)

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Presentación

En esta unidad continuamos trabajando con el CRUD de novedades. En este caso sumamos las funcionalidades necesarias para eliminar y modificar novedades desde nuestro administrador

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Objetivos

Que los participantes logren...

- Comprender el proceso de modificación de una novedad.

- Eliminar novedades de la base de datos.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Bloques temáticos

1. Eliminar novedades.
2. Modificar novedades

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

1. Eliminar novedades

Paso 1

En el archivo **models/novedades Model.js** agregaremos la función necesaria para eliminar la novedad según el id.

La función **delete Novedad ById** recibe como parámetro el id de la novedad y ejecuta el código necesario para eliminarla de la base de datos.

```
async function deleteNovedadById(id) {
    var query = "delete from novedades where id = ? ";
    var rows = await pool.query(query, [id]);
    return rows;
}

module.exports = { getNovedades, deleteNovedadById }
```

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Paso 2

En el archivo **routes/admin/novedades.js** agregaremos el controlador que será el encargado de capturar las rutas de eliminación y llamará a la función que acabamos de crear en el modelo, pasando como parámetro el valor que reciba por url.

```
router.get('/eliminar/:id', async (req, res, next) => {
  var id = req.params.id;
  await novedadesModel.deleteNovedadById(id);
  res.redirect('/admin/novedades')
});
```

Paso 3

Modificamos nuestro archivo **views/admin/novedades.hbs** e incluimos un link a nuestro nuevo controlador de eliminar.

```
<a href="/admin/novedades/eliminar/{{id}}"
class="btn btn-sm btn-secondary"><i class="fa fa-trash"></i></a>
```

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

1.Modificar novedades

Paso 1

Modificamos nuestro archivo **views/admin/novedades.hbs** e incluimos un link a la ruta de modificar.

```
<a href="/admin/novedades/modificar/{{id}}" class="btn btn-sm
btn-secondary"><i class="fa fa-pencil"></i></a>
```

Paso 2

En el archivo **models/novedades Model.js** agregaremos dos funciones nuevas. Una nos permitirá obtener una noticia única de la base de datos utilizando el id de la misma para seleccionarla. La otra será la encargada de modificar los campos de la novedad que seleccionemos por id y que reciba como parámetro.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

```
async function getNovedadById(id) {
    var query = "select * from novedades where id = ? ";
    var rows = await pool.query(query, [id]);
    return rows[0];
}

async function modificarNovedadById(obj, id) {
    try {
        var query = "update novedades set ? where id=?";
        var rows = await pool.query(query, [obj, id]);
        return rows;
    } catch (error) {
        throw error;
}

module.exports = { getNovedades, insertNovedad,
deleteNovedadById, getNovedadById, modificarNovedadById }
}
```

Paso 3

En el archivo **routes/admin/novedades.js** creamos el controlador de ruta necesario para imprimir el formulario de modificación. Esta ruta tiene la particularidad, al igual que la de eliminar, de recibir como parámetro el id de la novedad. Este id se utilizara para llamar a la función previamente creada y pasar la novedad seleccionada al template.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

```
router.get('/modificar/:id', async (req, res, next) => {

  let id = req.params.id;
  let novedad = await novedadesModel.getNovedadById(id);
  res.render('admin/modificar', {
    layout: 'admin/layout',
    novedad
  });
});
```

Paso 4

Creamos la vista **views/admin/modificar.hbs** donde armaremos el formulario para modificar. Este es muy similar al de agregar, con la salvedad de que los valores de los campos vienen predefinidos por los campos de la novedad que trajimos de la base de datos.

Además agregamos un campo oculto que utilizaremos en el envio del formulario para saber qué novedad debemos modificar.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

```
<div class="contanier" style="margin:50px auto">
    <div class="row">
        <div class="col-6 offset-3">
            <h4>Modificar la novedad</h4>
            <form method="post" action="/admin/novedades/modificar">
                <input type="hidden" value="{{novedad.id}}" name="id">
                <div class="form-group">
                    <input type="text" placeholder="Titulo"
                    value="{{novedad.titulo}}" class="form-control"
                    name="titulo">
                </div>
                <div class="form-group">
                    <textarea placeholder="Subtitulo" class="form-control"
                        name="subtitulo">{{novedad.subtitulo}}</textarea>
                </div>
                <div class="form-group">
                    <textarea placeholder="Cuerpo" class="form-control"
                    name="cuerpo">{{novedad.cuerpo}}</textarea>
                </div>
                <button type="submit" class="btn btn-primary">Modificar</button>
            </form>
        </div>
    </div>
</div>
```

Paso 5

En el archivo **routes/admin/novedades.js** creamos el controlador encargado de recibir los datos del formulario y pasarlos a la función de model para efectuar la modificación de la novedad en la base de datos.

En caso de éxito redirigimos al usuario al listado de novedades, de lo contrario enviamos una variable de error y el mensaje describiendo el mismo.

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

```
router.post('/modificar', async (req, res, next) => {
  try {
    let obj = {
      titulo: req.body.titulo,
      subtitulo: req.body.subtitulo,
      cuerpo: req.body.cuerpo
    }

    await novedadesModel.modificarNovedadById(obj, req.body.id);
    res.redirect('/admin/novedades');
  }
  catch (error) {
    console.log(error)
    res.render('admin/modificar', {
      layout: 'admin/layout',
      error: true, message: 'No se modifico la novedad'
    })
  }
})
```

Paso 6

En el archivo **views/admin/modificar.hbs** agregamos el if para que imprima el error en caso de haber uno.

```
{{#if error}}
    <h5>{{message}}</h5>
{{/if}}
```

Centro de e-Learning SCEU UTN - BA. Medrano 951 2do piso
(1179) // Tel. +54 11 7078- 8073 / Fax +54 11 4032 0148
www.sceu.frba.utn.edu.ar/e-learning

Bibliografía utilizada y sugerida

Artículos de revista en formato electrónico:

Handlebars. Disponible desde la URL: https://handlebarsjs.com/

npmjs. Disponible desde la URL: https://www.npmjs.com/