

```
1 import { check, validationResult } from 'express-validator'
2 import bcrypt from 'bcrypt';
3 import Usuario from '../models/Usuario.js';
4 import { generarId, generarJWT } from '../helpers/tokens.js'
5 import { emailRegistro, emailOlvidePassword } from '../helpers/emails.js'
6
7 const formularioRegistro = (req, res) => {
8   res.render('auth/registrarse', {
9     pagina : 'Crear Cuenta',
10     csrfToken: req.csrfToken()
11   });
12 }
```

ORM

Object Relational Mapper

```
25   csrfToken: req.csrfToken(),
26   errores : resultado.array(),
27   pagina : 'Crear Cuenta',
28   usuario : {
```

Juan Pablo De la torre Valdez

www.codigoconjuan.com



¿Qué es ORM?



ORM son las iniciales de Object Relational Mapping

Es una técnica que se utiliza donde los datos de una base de datos son tratados como Objetos, utilizando un paradigma de Programación Orientada a Objetos

Node.js tiene una gran cantidad de ORM's que se instalan como librería

En MVC; un ORM se relaciona bastante con el Modelo



Ventajas de un ORM



Comenzar a crear aplicaciones que utilicen bases de datos, sin necesidad de escribir código SQL y tampoco saber sobre modelado de bases de datos.

Velocidad en desarrollo ya que tienen una gran cantidad de métodos para crear, listar, actualizar o eliminar datos.

La mayoría cuentan con todas las medidas de seguridad.



Ejemplo

ORM



```
INSERT INTO `categorias` (`nombre`) VALUES ('Casa');
```

```
Categoria.create({ nombre: 'Casa' })
```



Ejemplos de ORM



Prisma

Mongoose

TypeORM

Sequelize

Bookshelf.js

