

# Repositorios MongoDB en Plugcore



Plug Soluciones TIC S.L.  
info@plugcore.com



<https://github.com/plugcore/plugcore>



<https://www.linkedin.com/company/plugcore>



<https://twitter.com/plugcoreit>

# Repositorios MongoDB

1. **Introducción**
2. **Ejemplo de servicio**



# Introducción

- Todo lo que vamos a comentar hoy está en el paquete [@plugcore/ds-mongodb](https://github.com/plugcore/ds-mongodb), con lo cual si no lo tenemos instalado lo vamos a necesitar.
- Esto es una clase de utilidad que hay dentro del datasource que nos ayudará con las operaciones más comunes de una collection y además tendremos la opción de agregar atributos de auditoría o de guardar los documentos eliminados de una colección en otra colección.
- Es muy parecido usar directamente la collection de MongoDB como hemos visto en un video anterior, pero agregando estas opciones de utilidad.



# Ejemplo de servicio

```
// Ejemplo de servicio que usa MongoDBRepository
@Service({ connection: 'mymongodb' })
export class TagRepository implements OnInit {

    // Aquí definimos nuestro MongoDBRepository que guardará documentos de tipo
    "PostModel" y usará el atributo "id" como identificador.
    private repository: MongoDBRepository<PostModel, 'id'>;

    constructor(
        private mongoDbDatasource: MongoDBDatasource
    ) {}
```



```
public async onInit() {  
  
    this.repository = new MongoDBRepository<PostModel, 'id'>({  
        dataSource: this.mongodbDatasource,  
        collectionName: 'tags', // Nombre de la colección en MongoDB  
        idAttribute: 'id', // Tag id  
        auditEditedEntries: true, // Añade los campos de creación y actualización a  
los documentos editados  
        auditDeletedEntries: true, // Todos los documentos que se eliminen se  
guardarán en una colección a parte  
    });  
  
    await this.repository.onInit(); // Tenemos que ejecutar el init para que se  
conecte a la collection de MongoDB  
  
}
```



```
// Ejemplo de operaciones CRUD básicas
public async create(Post: PostModel) {
    return this.repository.insertOne(Post);
}

public async findById(id: string): Promise<PostModel | null> {
    return this.repository.findById(id);
}

public async update(Post: PostModel) {
    return this.repository.updateOne({ id: Post.id }, Post);
}

public async deleteOne(id: PostModel['id']) {
    return this.repository.deleteOne({ id });
}
```



# GRACIAS POR SU TIEMPO

Para más información

[info@plugcore.com](mailto:info@plugcore.com)

[germanml@plugcore.com](mailto:germanml@plugcore.com)

[sergiolc@plugcore.com](mailto:sergiolc@plugcore.com)



Diario de Mallorca

Última Hora

#EMPRENBIT

