



Acceso a Datos

UT04- BBDD NoSQL

Act.02 - Blog Noticias

UT04. Act01

CRUD MongoDB

Objetivo

El objetivo de este ejercicio es diseñar y desarrollar un sistema básico de gestión de un blog de noticias utilizando Kotlin y MongoDB como base de datos. Este ejercicio permitirá poner en práctica conceptos de modelado de datos, programación orientada a objetos, y operaciones básicas de persistencia en bases de datos documentales.

Especificaciones

Se necesita diseñar un blog de noticias donde los usuarios registrados puedan publicar noticias y los visitantes puedan escribir comentarios sobre dichas noticias. A continuación, se detallan los requisitos del sistema:

1. Gestión de usuarios

Cada usuario registrado debe contar con los siguientes datos:

- `_id` : El email del usuario
- Nombre completo.
- Nombre de usuario único.
- Estado (banned / not banned, activo / inactivo).
- Dirección postal: calle, número, puerta, código postal y ciudad.
- Teléfonos de contacto (pueden tener varios).

2. Gestión de noticias

Cada noticia debe contener:

- Un título.
- Un cuerpo.
- Una fecha de publicación.
- Un autor (el usuario que publicó la noticia).
- Una lista opcional de etiquetas (tags).

3. Gestión de comentarios

Cada noticia debe permitir recibir comentarios. Cada comentario debe incluir:

- El nombre del usuario que escribió el comentario.
- La noticia donde se escribió el comentario

- El texto del comentario.
- La fecha y hora en la que fue escrito.

4. Requisitos funcionales

El programa debe ser capaz de realizar las siguientes operaciones:

- Los usuarios registrados pueden publicar noticias
- Los usuarios pueden escribir comentarios (a no ser que estén en estado inactivo/banned)
- Registrar usuarios.
- Publicar noticias.
- Listar las noticias publicadas por un usuario.
- Listar los comentarios de una noticia.
- Buscar noticias por etiquetas.
- Listar las 10 últimas noticias publicadas

5. Lógica de negocio

- El email del usuario y el nick deben ser únicos
- La fecha de publicación de una noticia debe ser única y no editable.
- Los comentarios son inmutables una vez publicados.

Entrega

1. Código fuente de la aplicación.
2. Documento explicativo que incluya:
 - Diagrama de clases.
 - Explicación del diseño y las decisiones tomadas.
 - Capturas de pantalla o logs que demuestren el correcto funcionamiento de la aplicación.

RAs y CEs evaluados

RA 5. Desarrolla aplicaciones que gestionan la información almacenada en bases de datos nativas XML evaluando y utilizando clases específicas.

a) Se han valorado las ventajas e inconvenientes de utilizar una base de datos nativa XML.

Indicadores	Niveles de logro		
	Bien	Regular	Insuficiente
Explicación del diseño y las decisiones tomadas	100% Explica de forma adecuada la elección del diseño de la BDD	50%	0%

b) Se ha instalado el gestor de base de datos.

c) Se ha configurado el gestor de base de datos.

d) Se ha establecido la conexión con la base de datos.

Indicadores	Niveles de logro		
	Bien	Regular	Insuficiente
Realiza la configuración básica de una base de datos documental	100% Realiza la configuración de la base de datos documental, aplicando patrones de diseño para garantizar una correcta conexión a la BDD. Cierra la conexión cuando	50%	0%

	hay excepciones o termina las operaciones.		
--	--	--	--

e) Se han desarrollado aplicaciones que efectúan consultas sobre el contenido de la base de datos.

Indicadores	Niveles de logro		
	Bien	Regular	Insuficiente
Realiza consultas sobre la base de datos NoSQL	100% Cumple con todas las funcionalidades que se exigen	50%	0%

f) Se han añadido y eliminado colecciones de la base de datos.

Indicadores	Niveles de logro		
	Bien	Regular	Insuficiente
Realiza inserción y eliminación de documentos	100% Cumple con todas las funcionalidades que se exigen	50%	0%

g) Se han desarrollado aplicaciones para añadir, modificar y eliminar documentos XML de la base de datos.

Indicadores	Niveles de logro		
	Bien	Regular	Insuficiente
	100%	50%	0%

Desarrolla una aplicación que gestiona una base de datos documental	Cumple con todos los requisitos que se exigen		
	100% El documento entregable abarca todos los puntos		
	La aplicación se divide en clases, separando la lógica correctamente		