



AE U2.6 - DESPLIEGUE EN MONGODB + U2.7 ROOTER



Nombre y apellidos: Cristina Sandoval
Laborde

Curso: 2ºDAM

Asignatura: Optativa GS

Índice

| | |
|---|----|
| 1.Introducción..... | 1 |
| 2.Entidades principales | 2 |
| 3.Relaciones en MongoDB..... | 5 |
| 4.Implementación en MongoDB | 6 |
| 5.Cambios generales al pasar de SQL a MongoDB | 13 |

1. Introducción

StudyMatchKey es una aplicación educativa diseñada para ayudar a los estudiantes de Bachillerato a descubrir y aplicar el método de estudio que mejor se adapta a sus hábitos y preferencias. A través de un test inicial, la app identifica el estilo de aprendizaje predominante del usuario y ofrece recomendaciones personalizadas, como el uso de mapas mentales, la técnica Pomodoro, flashcards o el método Feynman.

Además, la aplicación integra un calendario inteligente para organizar tareas y exámenes, un asistente paso a paso que guía en la aplicación de técnicas activas, y un sistema de gestión de materiales que centraliza apuntes y recursos. Todo ello se complementa con notificaciones automáticas y un enfoque en la privacidad y seguridad de los datos.

El objetivo principal de StudyMatchKey es **mejorar la organización, la motivación y el rendimiento académico**, reduciendo el estrés y el riesgo de fracaso escolar mediante un acompañamiento visual, práctico y adaptado a cada estudiante.

Para el desarrollo de **StudyMatchKey** se ha optado por utilizar **MongoDB**, una base de datos no relacional orientada a documentos. La elección se fundamenta en varias razones:

- **Flexibilidad en el almacenamiento de datos:** los perfiles de usuario, resultados de test y materiales de estudio no siguen una estructura fija. MongoDB permite guardar información en formato JSON dinámico, adaptándose fácilmente a cambios futuros sin necesidad de modificar esquemas rígidos.
- **Escalabilidad:** al tratarse de una aplicación que puede crecer en número de usuarios y volumen de datos (tareas, exámenes, materiales, notificaciones), MongoDB ofrece una arquitectura distribuida que facilita el escalado horizontal.
- **Integración con aplicaciones modernas:** al ser no relacional, se adapta mejor a entornos de desarrollo ágiles y aplicaciones móviles, donde los datos cambian con frecuencia y se necesitan respuestas inmediatas.
- **Gestión de datos heterogéneos:** StudyMatchKey maneja distintos tipos de información (test, calendario, materiales, notificaciones). Una base no relacional permite almacenar todo en colecciones relacionadas sin necesidad de normalizar excesivamente como en SQL.

2.Entidades principales

Usuario

Atributos:

- ID_Usuario
- ID_Entidad
- Nombre
- Fecha_De_Nacimiento
- Email
- Contraseña
- Fecha_De_Registro

Descripción: Representa a cada estudiante dentro de la aplicación. Incluye datos mínimos necesarios para identificar al usuario y proteger su cuenta.

Calendario

Atributos:

- ID_Calendario
- ID_Usuario
- Fecha_Evento
- Tipo_De_Evento
- Descripcion_Evento

Descripción: Permite gestionar las tareas académicas del usuario y los recordatorios automáticos.

TecnicaEstudio

Atributos:

- ID_Tecnica
- Nombre_Tecnica
- Descripcion_TecnicaEstudio
- Tipo_TecnicaEstudio

Descripción: Conjunto de técnicas sugeridas por la aplicación, útiles para generar recomendaciones personalizadas.

MetodoEstudio

Atributos:

- ID_Metodo
- Nombre_Metodo
- Descripcion_Metodo
- Categoria

Descripción: Conjunto de métodos sugeridos por la aplicación. Se asignan a estudiantes dependiendo de la asignatura y sus preferencias.

Recomendacion

Atributos:

- ID_Recomendacion
- ID_Usuario
- ID_Tecnica
- Fecha_Recomendacion
- Efectividad

Descripción: Registra qué técnica se recomienda a cada usuario y cómo ha funcionado en la práctica.

ArchivoEstudio

Atributos:

- ID_Archivo
- ID_Usuario
- Tipo_De_Archivo
- Nombre_Archivo
- Ruta_De_Almacenamiento
- Fecha_De_Subida

Descripción: Material de estudio subido por el usuario. Se almacena en espacios seguros y cifrados.

MetricasEstudio

Atributos:

- ID_Metrica
- ID_Usuario
- Horas_De_Estudio
- Tiempo_De_Concentración
- Uso_De_Tecnicas
- Fecha_Registro

Descripción: Representa las estadísticas que usa la app para personalizar las recomendaciones y mostrar progreso.

Entidad / Organismo

Atributos:

- ID_Entidad
- Nombre_Entidad
- Tipo
- Ubicacion

Descripción: Centro educativo, institución, organismo o academia a la que pertenece el estudiante. Sirve para organizar usuarios por colegios,institutos, universidad, etc.

Asignatura

Atributos:

- ID_Asignatura
- ID_Entidad
- Nombre_Asignatura

Descripción: Materias impartidas por la entidad educativa. Sirven para asociar usuarios con sus estudios y generar recomendaciones según el tipo de asignatura.

3.Relaciones en MongoDB

Usuario — Entidad

Tipo de relación: (N : 1)

Explicación (MongoDB):

- Muchos usuarios pueden pertenecer a la misma entidad.
- Cada usuario guarda **solo el ID de la entidad**.
- No se usa una "tabla intermedia" porque **MongoDB no lo necesita**.

Entidad — Asignatura

Tipo de relación: (1 : N)

(Una Entidad imparte muchas Asignaturas)

Explicación (MongoDB):

- Una entidad ofrece muchas asignaturas.
- Cada asignatura solo referencia la entidad.
- No se embeben asignaturas dentro de entidad porque pueden ser cientos.

Usuario — Asignatura

Tipo de relación: (N : M)

Un usuario puede matricularse en muchas asignaturas.

Una asignatura puede tener muchos usuarios matriculados.

En MongoDB se representa con **un array dentro del Usuario**

Explicación (MongoDB):

- Un usuario puede tener varias asignaturas → se ponen en un array.
- Una asignatura tiene varios usuarios → **MongoDB no necesita representarlo**.
- No se usa tabla intermedia

Asignatura — MétodoEstudio

Tipo de relación: (N : M)

La relación se gestiona **cuando el usuario recibe un método**, no antes.

Usuario — MétodoEstudio (por asignatura)

Tipo de relación: (**N : M**)

Explicación (MongoDB):

- Un usuario puede tener muchos métodos.
- Cada asignatura puede estar asociada a diferentes métodos dependiendo del perfil del usuario.
- No es que el método dependa de la asignatura, sino que la combinación Usuario + Asignatura determina qué métodos se recomiendan, es decir, el perfil del usuario dentro de esa asignatura concreta

Usuario — Calendario

Tipo de relación: (**1 : N**)

Explicación (MongoDB):

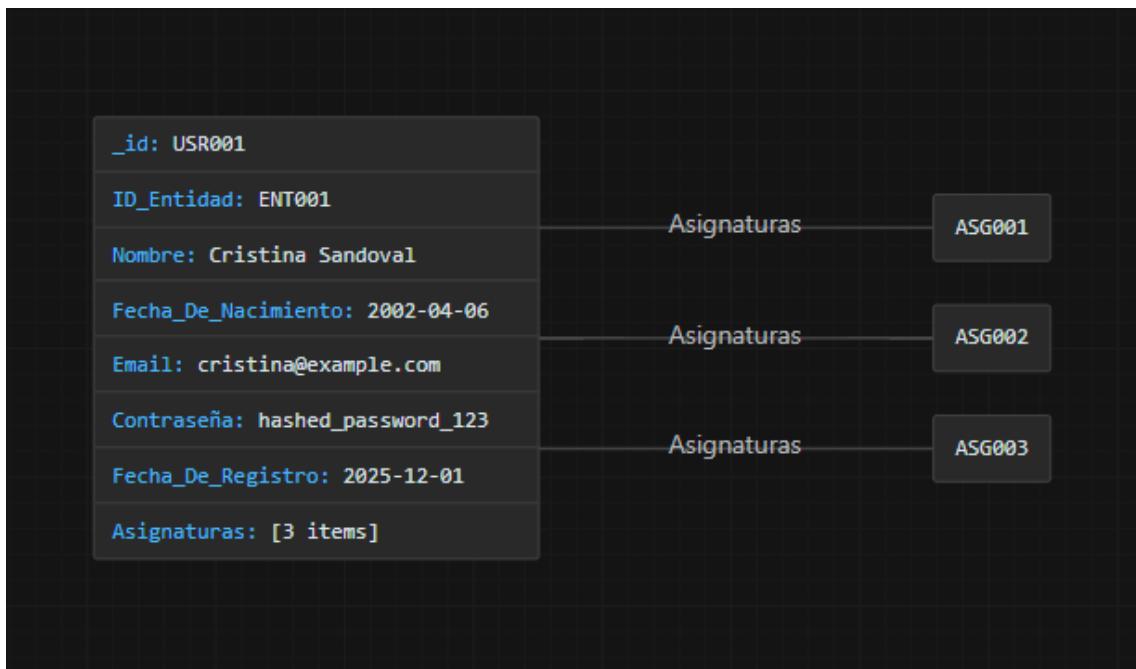
- Un usuario puede tener muchos eventos en su calendario.
- Se usa referencia por ID.

4. Implementación en MongoDB

Colección Usuario: relación con Entidad (ID_Entidad) y el array con asignaturas (N:M)

```
{  
    "_id": "USR001",  
    "ID_Entidad": "ENT001",  
    "Nombre": "Cristina Sandoval",  
    "Fecha_De_Nacimiento": "2002-04-06",  
    "Email": "cristina@example.com",  
    "Contraseña": "hashed_password_123",  
    "Fecha_De_Registro": "2025-12-01",  
    "Asignaturas": ["ASG001", "ASG002", "ASG003"]  
}
```

```
{  
    "_id": "USR001",  
    "ID_Entidad": "ENT001",  
    "Nombre": "Cristina Sandoval",  
    "Fecha_De_Nacimiento": "2002-04-06",  
    "Email": "cristina@example.com",  
    "Contraseña": "hashed_password_123",  
    "Fecha_De_Registro": "2025-12-01",  
    "Asignaturas": [  
        "ASG001",  
        "ASG002",  
        "ASG003"  
    ]  
}
```



Colección: Entidad / Organismo

```
{  
    "_id": "ENT001",  
    "Nombre_Entidad": "IES Tecnológico Málaga",  
    "Tipo": "Instituto",  
    "Ubicacion": "Málaga, España"  
}
```

```
{  
    "_id": "ENT001",  
    "Nombre_Entidad": "IES Tecnológico Málaga",  
    "Tipo": "Instituto",  
    "Ubicacion": "Málaga, España"  
}
```

| |
|--|
| _id: ENT001 |
| Nombre_Entidad: IES Tecnológico Málaga |
| Tipo: Instituto |
| Ubicacion: Málaga, España |

Colección: Asignatura

Relacionada con la entidad.

```
{  
    "_id": "ASG001",  
    "ID_Entidad": "ENT001",  
    "Nombre_Asignatura": "Programación"  
}
```

```
{  
    "_id": "ASG001",  
    "ID_Entidad": "ENT001",  
    "Nombre_Asignatura": "Programación"  
}
```

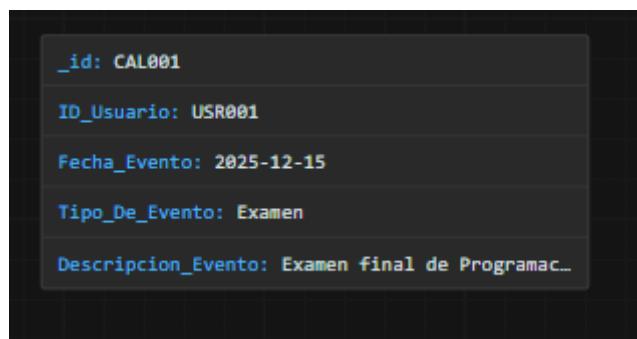
| |
|---------------------------------|
| _id: ASG001 |
| ID_Entidad: ENT001 |
| Nombre_Asignatura: Programación |

Colección: Calendario

Eventos propios de cada usuario

```
{  
    "_id": "CAL001",  
    "ID_Usuario": "USR001",  
    "Fecha_Evento": "2025-12-15",  
    "Tipo_De_Evento": "Examen",  
    "Descripcion_Evento": "Examen final de Programación"  
}
```

```
{  
    "_id": "CAL001",  
    "ID_Usuario": "USR001",  
    "Fecha_Evento": "2025-12-15",  
    "Tipo_De_Evento": "Examen",  
    "Descripcion_Evento": "Examen final de Programación"  
}
```



Colección: TecnicaEstudio

```
{  
    "_id": "TEC001",  
    "Nombre_Tecnica": "Pomodoro",  
    "Descripcion_TecnicaEstudio": "Estudiar en intervalos de 25 minutos con 5 minutos  
de descanso.",  
    "Tipo_TecnicaEstudio": "Concentración"  
}
```

```
{  
    "_id": "TEC001",  
    "Nombre_Tecnica": "Pomodoro",  
    "Descripcion_TecnicaEstudio": "Estudiar en intervalos de 25 minutos con 5 minutos de descanso.",  
    "Tipo_TecnicaEstudio": "Concentración"  
}
```

```
_id: TEC001  
Nombre_Tecnica: Pomodoro  
Descripcion_TecnicaEstudio: Estudiar en intervalos de 25 minu...  
Tipo_TecnicaEstudio: Concentración
```

Colección: MetodoEstudio

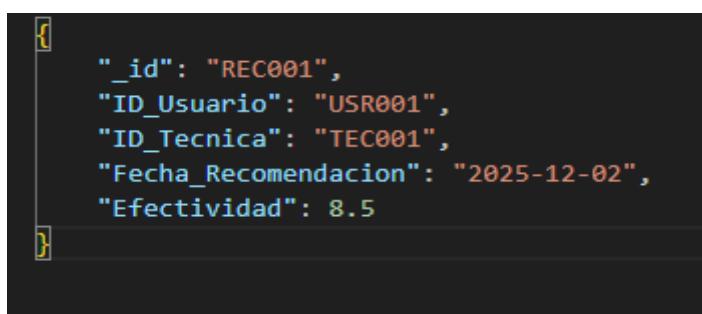
```
{  
    "_id": "MET001",  
    "Nombre_Metodo": "Mapas Mentales",  
    "Descripcion_Metodo": "Organización gráfica del contenido.",  
    "Categoria": "Visual"  
}
```

```
{  
    "_id": "MET001",  
    "Nombre_Metodo": "Mapas Mentales",  
    "Descripcion_Metodo": "Organización gráfica del contenido.",  
    "Categoria": "Visual"  
}
```

```
_id: MET001  
Nombre_Metodo: Mapas Mentales  
Descripcion_Metodo: Organización gráfica del ...  
Categoria: Visual
```

Colección: Recomendación

```
{  
    "_id": "REC001",  
    "ID_Usuario": "USR001",  
    "ID_Tecnica": "TEC001",  
    "Fecha_Recomendacion": "2025-12-02",  
    "Efectividad": 8.5  
}
```



```
{  
    "_id": "REC001",  
    "ID_Usuario": "USR001",  
    "ID_Tecnica": "TEC001",  
    "Fecha_Recomendacion": "2025-12-02",  
    "Efectividad": 8.5  
}
```



```
_id: REC001  
ID_Usuario: USR001  
ID_Tecnica: TEC001  
Fecha_Recomendacion: 2025-12-02  
Efectividad: 8.5
```

Colección: ArchivoEstudio

```
{  
    "_id": "ARC001",  
    "ID_Usuario": "USR001",  
    "Tipo_De_Archivo": "PDF",  
    "Nombre_Archivo": "apuntes_programacion.pdf",  
    "Ruta_De_Almacenamiento": "/uploads/USR001/apuntes_programacion.pdf",  
    "Fecha_De_Subida": "2025-12-01"  
}
```

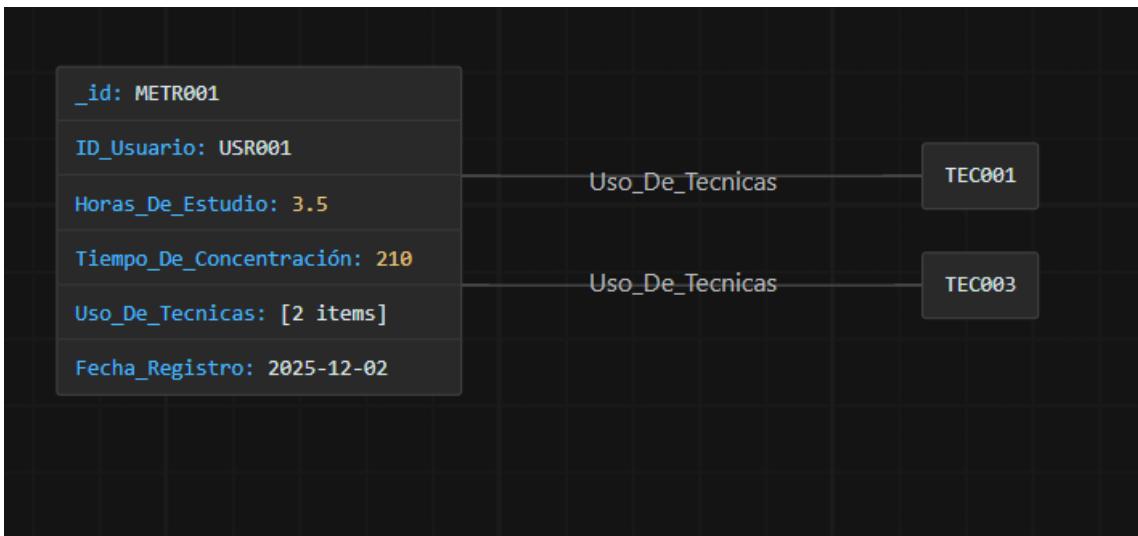
```
{  
    "_id": "ARC001",  
    "ID_Usuario": "USR001",  
    "Tipo_De_Archivo": "PDF",  
    "Nombre_Archivo": "apuntes_programacion.pdf",  
    "Ruta_De_Almacenamiento": "/uploads/USR001/apuntes_programacion.pdf",  
    "Fecha_De_Subida": "2025-12-01"  
}
```

```
_id: ARC001  
ID_Usuario: USR001  
Tipo_De_Archivo: PDF  
Nombre_Archivo: apuntes_programacion.pdf  
Ruta_De_Almacenamiento: /uploads/USR001/apunt...  
Fecha_De_Subida: 2025-12-01
```

Colección: MetricasEstudio

```
{  
    "_id": "METR001",  
    "ID_Usuario": "USR001",  
    "Horas_De_Estudio": 3.5,  
    "Tiempo_De_Concentración": 210,  
    "Uso_De_Tecnicas": ["TEC001", "TEC003"],  
    "Fecha_Registro": "2025-12-02"  
}
```

```
{
    "_id": "METR001",
    "ID_Usuario": "USR001",
    "Horas_De_Estudio": 3.5,
    "Tiempo_De_Concentración": 210,
    "Uso_De_Técnicas": [
        "TEC001",
        "TEC003"
    ],
    "Fecha_Registro": "2025-12-02"
}
```



5.Cambios generales al pasar de SQL a MongoDB

En el modelo relacional original existían:

- Tablas independientes con claves primarias .
- Relaciones 1:N mediante claves foráneas.
- Relaciones N:M mediante tablas intermedias (Usuario_Asignatura, Asignatura_Metodo, etc.).
- Reglas de integridad referencial estrictas.

MongoDB trabaja con documentos y colecciones, por lo que fue necesario:

- Eliminar todas las tablas intermedias

Ya no son necesarias gracias a los arrays dentro de los documentos.

-Reemplazar claves foráneas por referencias simples (ID_Entidad, ID_Usuario, etc.)

Porque MongoDB no usa FOREIGN KEY ni JOIN.

- Mantener cada entidad como colección separada

Para asegurar independencia, escalabilidad y claridad del modelo.

5.1. Explicación de los cambios

✓ Relación Usuario — Entidad (N:1)

Antes (SQL):

- Usuario tenía una clave foránea ID_Entidad.

Ahora (MongoDB):

- Se mantiene igual, pero como referencia simple.
- No se necesitan restricciones ni validaciones automáticas.

Motivo del cambio:

MongoDB no usa claves foráneas; por ello, **se conserva solo el identificador como referencia.**

✓ Relación Entidad — Asignatura (1:N)

Antes (SQL):

- Asignatura tenía ID_Entidad como clave foránea.

Ahora (MongoDB):

- Exactamente igual: se mantiene un atributo ID_Entidad dentro de cada asignatura.

Motivo:

Las asignaturas son muchas, por lo que **no deben ir dentro de una entidad.**

✓ Relación Usuario — Asignatura (N:M)

Este es uno de los cambios más importantes.

Antes (SQL):

- Existía una tabla intermedia Usuario_Asignatura.

Ahora (MongoDB):

- La tabla intermedia **desaparece**.
- En Usuario se añade un array:

"Asignaturas": ["ASG001", "ASG002", "ASG003"]

Motivo:

- MongoDB permite representar N:M con arrays.
- Se optimiza la operación principal: consultar asignaturas del usuario.

✓ **Relación Asignatura — MétodoEstudio (N:M)**

Antes (SQL):

- Tabla intermedia que conectaba asignaturas con métodos.

Ahora (MongoDB):

- **No se modela directamente.**
- Solo aparece la relación cuando un usuario recibe una recomendación.

Motivo:

La relación no es global, sino dependiente del perfil del usuario → por eso solo existe en la colección Recomendacion.

✓ **Relación Usuario — MétodoEstudio (N:M por asignatura)**

Antes (SQL):

- Requería tabla intermedia con usuario + asignatura + método.

Ahora (MongoDB):

- Se registra únicamente en la colección Recomendacion.
- No existe una tabla intermedia.

Motivo:

La recomendación depende del contexto del usuario(perfil), no de la asignatura global.

✓ **Relación Usuario — Calendario (1:N)**

Antes (SQL):

- Calendario tenía clave foránea a Usuario.

Ahora (MongoDB):

- Se mantiene igual, con ID_Usuario.

Motivo:

Cada usuario puede tener muchos eventos y es mejor una colección independiente.

Tablas eliminadas en la migración

Las siguientes tablas del modelo relacional ya no existen en MongoDB:

| Tabla SQL | Motivo de eliminación |
|---------------------------|---|
| Usuario_Asignatura | Sustituida por un array dentro de Usuario |
| Asignatura_Metodo | Relación implícita en Recomendacion |
| Usuario_Metodo | Ahora asociado mediante Recomendacion |

Estas eliminaciones son una consecuencia directa de la flexibilidad del modelo documental.

Enlace a repositorio Optativa → <https://github.com/crisanlab94/2-Dam/tree/main/Optativa>