

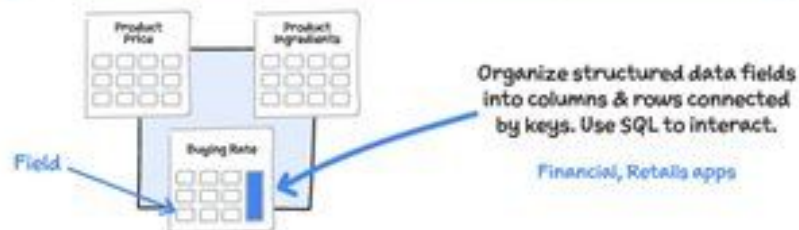
Introducción a MongoDB



Introduction to [Databases]

@pvergadia

RELATIONAL DATABASES



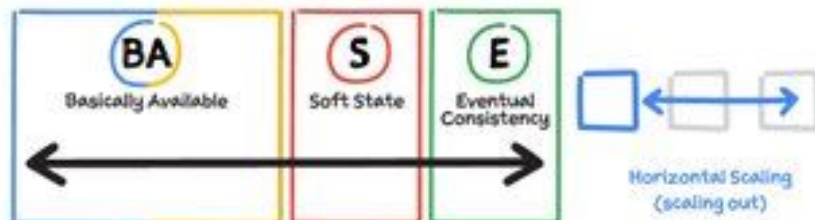
Use when...

- Data doesn't change very often
- Accuracy is crucial
- Transactional use cases (OLTP) & general purpose



NON RELATIONAL DATABASES

Organize unstructured data into any of these forms:



Works best when...

- Frequent changes in data
- Scale & availability is more crucial
- Analytics use cases (OLAP) & general purpose



Definición

MongoDB es una base de datos NoSQL open source orientada a almacenar documentos en formato **JSON** en **colecciones**. Debido a su arquitectura, permite escalar el sistema para almacenar grandes cantidades de datos.

En esta base de datos el esquema de los datos no es fijo y es posible modificar y añadir las propiedades de los documentos. Esto hace que sea muy flexible al almacenar diferentes tipos de documentos en las colecciones.

Las bases de datos en MongoDB almacenan **colecciones**, que son agrupaciones de **documentos** (las colecciones son equivalentes a las tablas en un modelo relacional). Cada documento puede ser único e independiente, con su propia estructura y campos. A cada documento se le asigna un identificador único en la colección llamado «**_id**». Los **documentos** a su vez contienen campos, que serían equivalentes a las columnas en un modelo relacional.

```
1 | {  
2 |   Nombre: Oscar,  
3 |   Ciudad: Madrid,  
4 |   Edad: 25  
5 | }
```

Definición

Strict schema vs. flexible data model

Relational Schema

Thing

id	title	description
12345-SFRobot-001	My Robot	Can do almost everything
67890-SFOven-1234	My Oven	Not to be used for bread

Property

id	thing_id	name	readOnly	writeOnly
4711-p1	67890-SFOven-1234	status	false	false

Action

id	thing_id	name	safe	idempotent
4711-a1	167890-SFOven-1234	toggle	false	false

Event

id	thing_id	name	readOnly	writeOnly
4711-e1	67890-SFOven-1234	overheating	false	false

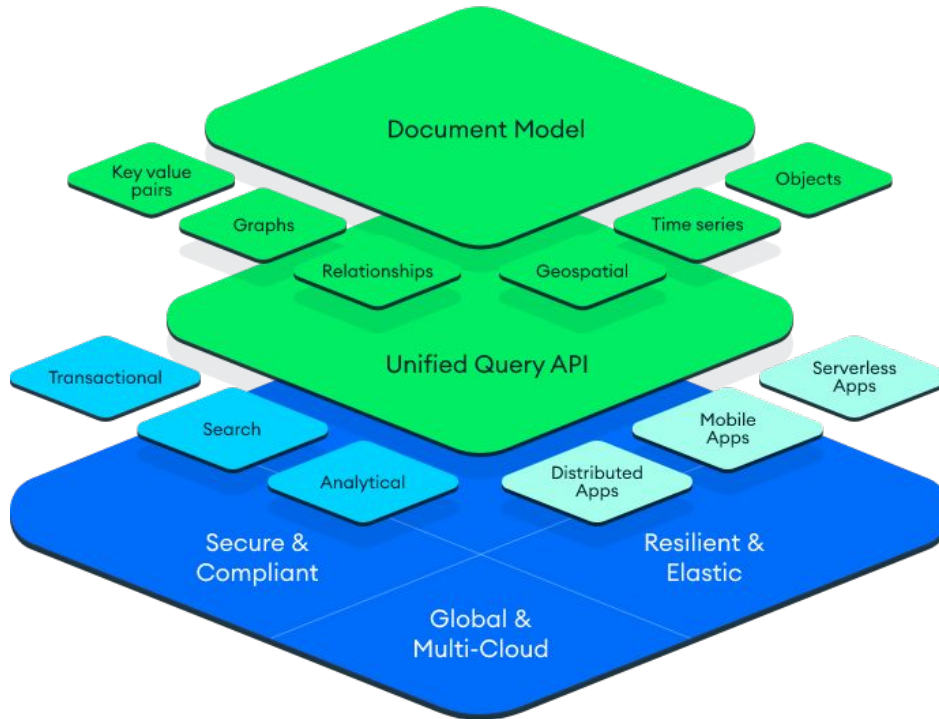
VS.



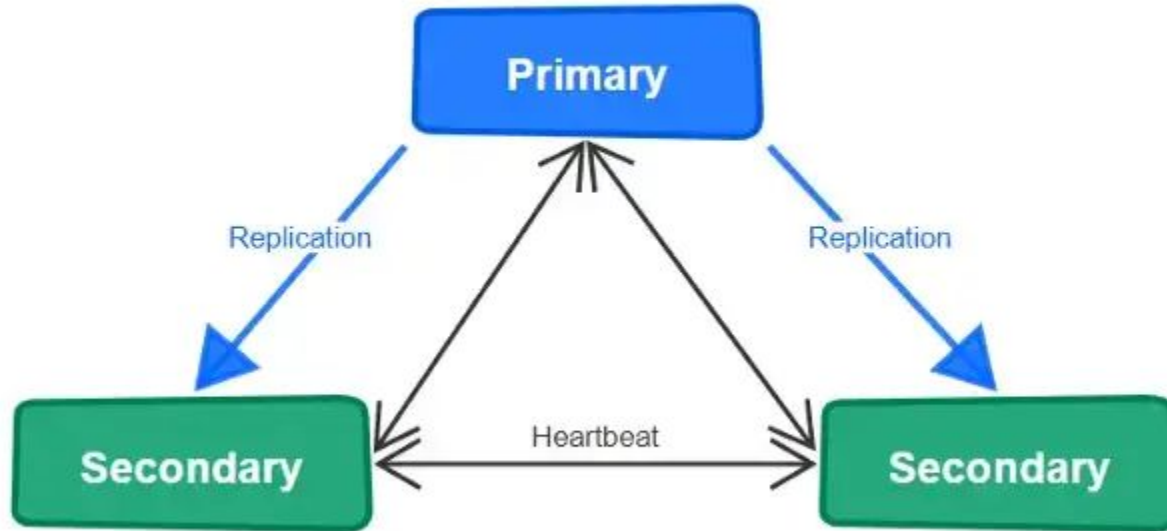
Document Model

```
{
  "id": "123456-WoTLamp-1234",
  "title": "My Lamp",
  "description": "A lamp in the room",
  "properties": {
    "status": {
      "type": "string",
      "readOnly" : false,
      "writeOnly" : false
    }
  },
  "actions": {
    "toggle": {
      "safe": false,
      "idempotent": false
    }
  },
  "events": {
    "overheating": {
      "data": {
        "type": "string",
        "readOnly" : false,
        "writeOnly" : false
      }
    }
  }
}
```

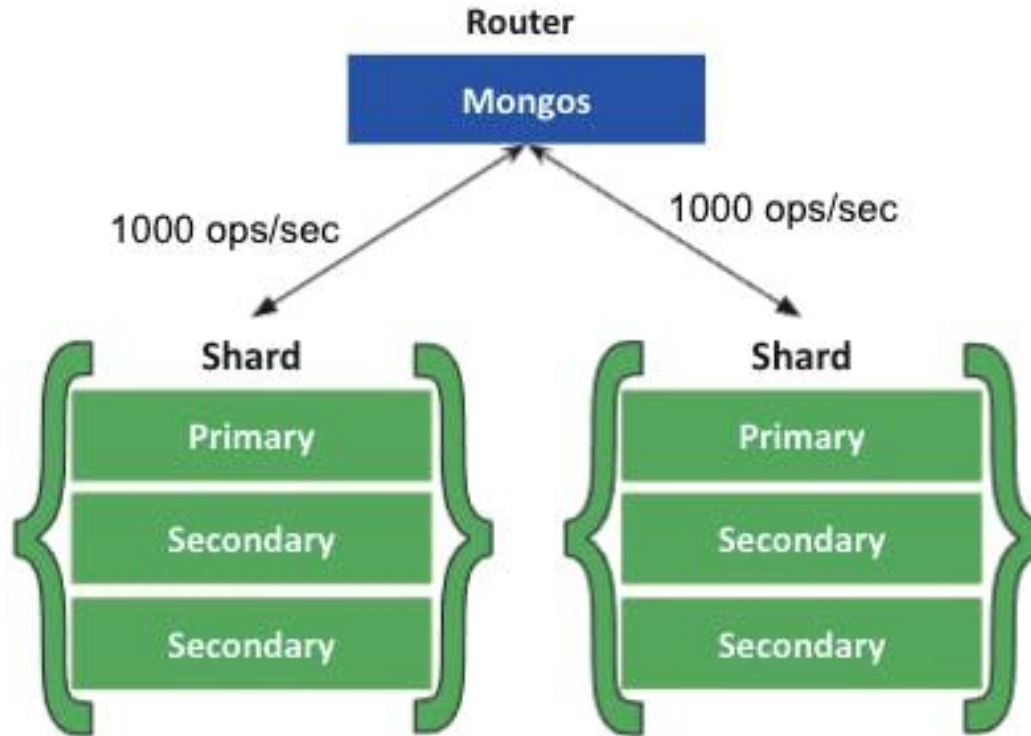
Arquitectura - Evolución



Arquitectura - Versión 6 - Primary mongod



Arquitectura - Versión 6 - Sharded (fragmentación) mongos



Modelado en MongoDB

- Relaciones 1:1

```
Order={
  "_id": "2934f",
  "salesDate": "2022-05-02",
  "customer":{
    "name": "Jack Beanstalk",
    "gender": "M",
    "rewardsMember": "True"
  }
}
```


Modelado en MongoDB

- Relaciones 1:N

Referencia al id

```
// patron document
{
  _id: "joe",
  name: "Joe Bookreader"
}

// address documents
{
  patron_id: "joe", // reference to patron document
  street: "123 Fake Street",
  city: "Faketon",
  state: "MA",
  zip: "12345"
}

{
  patron_id: "joe",
  street: "1 Some Other Street",
  city: "Boston",
  state: "MA",
  zip: "12345"
}
```

Modelado en MongoDB

- Relaciones 1:N

Por referencia en array

```
{
  name: "O'Reilly Media",
  founded: 1980,
  location: "CA",
  books: [123456789, 234567890, ...]
}

{
  _id: 123456789,
  title: "MongoDB: The Definitive Guide",
  author: [ "Kristina Chodorow", "Mike Dirolf" ],
  published_date: ISODate("2010-09-24"),
  pages: 216,
  language: "English"
}

{
  _id: 234567890,
  title: "50 Tips and Tricks for MongoDB Developer",
  author: "Kristina Chodorow",
  published_date: ISODate("2011-05-06"),
  pages: 68,
  language: "English"
}
```

Modelado en MongoDB

- Relaciones 1:N

Embedded...

Optimizado para un
determinado

atributo

MUY POTENTE, pero

MUY COSTOSO

```
{
  "_id": "joe",
  "name": "Joe Bookreader",
  "addresses": [
    {
      "street": "123 Fake Street",
      "city": "Faketon",
      "state": "MA",
      "zip": "12345"
    },
    {
      "street": "1 Some Other Street",
      "city": "Boston",
      "state": "MA",
      "zip": "12345"
    }
  ]
}
```

Modelado en MongoDB

- Relaciones N:M

```
Order={
  "_id": "2934f",
  "salesDate": "2022-05-02",
  "customer":{
    "name": "Jack Beanstalk",
    "gender": "M",
    "rewardsMember": "True"
  }
}
```

```
Item={
  "_id": "12",
  "name": "Pothos",
  "price":{
    "$numberDecimal": "8.00"
  },
  "orders": [ "2934f", "1b2df", "43de9" ]
}
```

Con un modelado optimizado para...

- Operaciones atómicas
- Búsquedas de palabras (keyboard search)
- Versionado de datos
- Información monetaria
- IoT (Sensores - Internet de las Cosas)
- Datos (pre)calculados

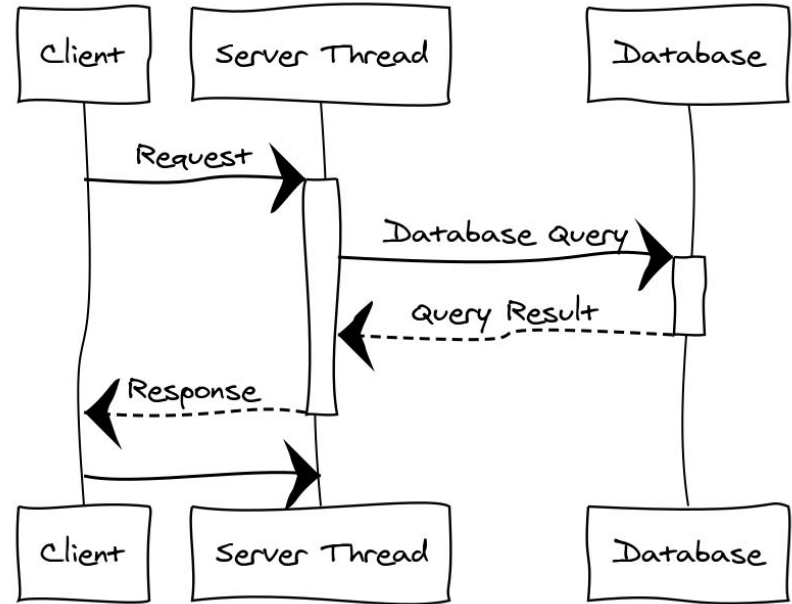
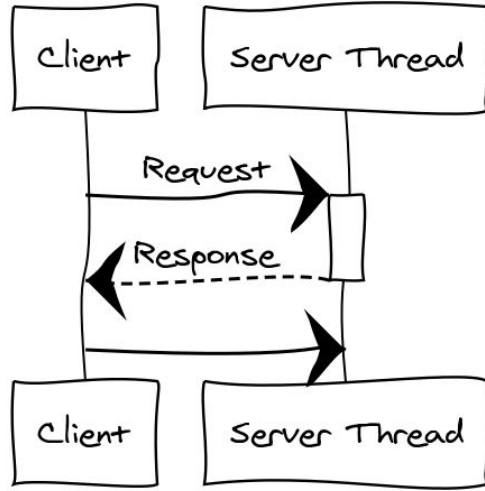




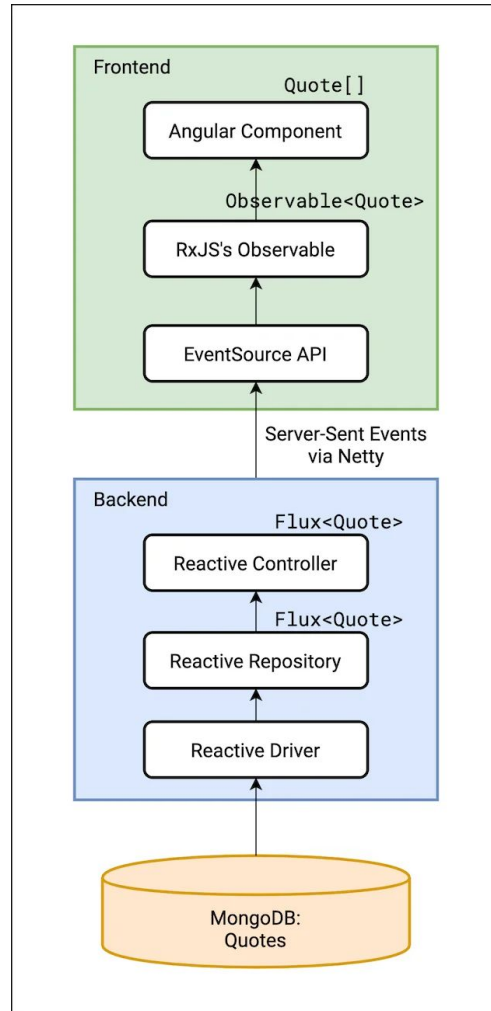
Más características

- Delegar en el motor de BBDD para las operaciones de consulta más potentes
 - API muy potente(y bien documentada)
- Apoyarse en el concepto de **pipeline**
 - Buscando la optimización
- Uso de Índices
 - Mejorando sensiblemente la performance
 - Apoyándose en las bondades del hardware
- Uso de Vistas Materializadas
 - Fundamentales al aumentar la volumetría
 - Herramienta potente y optimizada en MongoDB para su administración

Driver Reactivo

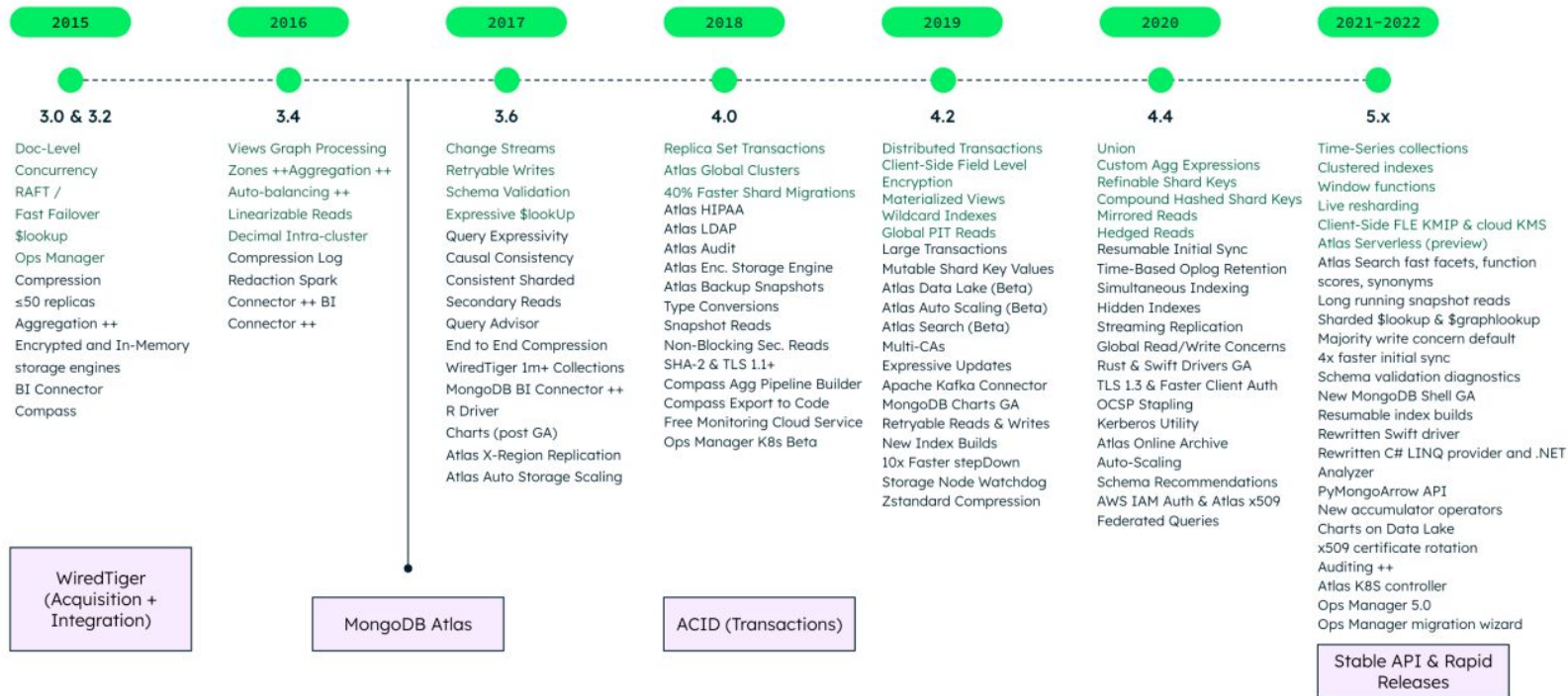


Driver Reactivo



Una evolución inteligente

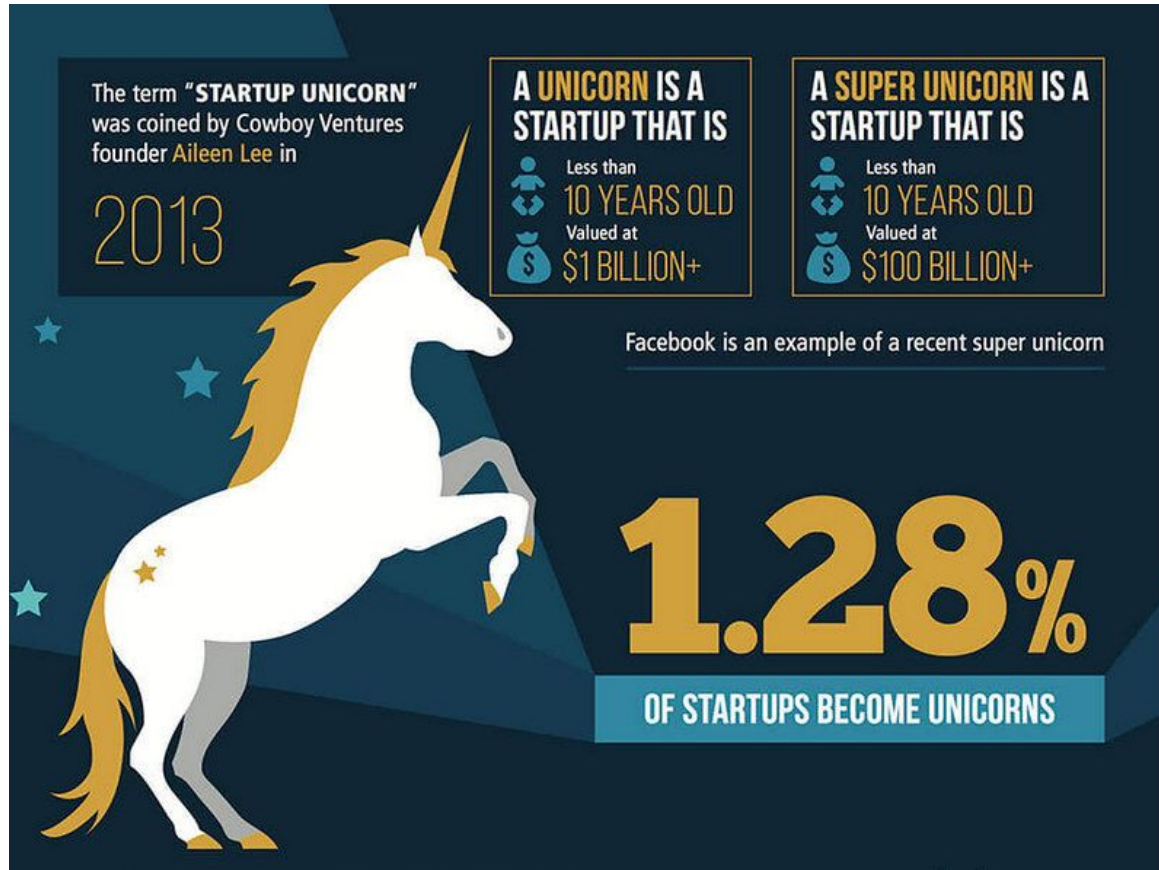
STACK OVERFLOW SURVEY | MOST WANTED DATABASE



Modelo de Negocio

- Software “Libre” con servicios de soporte de pago.
 - Precios pseudo públicos → [Enlace](#)
 - Orientado al Cloud. Pago por Uso
- Buscando atraer a la mayor cantidad de Comunidad posible
 - Soporte a muchos lenguajes
 - Atado a las tendencias
 - **Continuas** versiones *major* incorporando interesantes capacidades
- Partners con gran variedad de empresas y organismos públicos
 - Internacionales → Amazon, Google, Docker
 - Locales → Indra, Atos, Accenture, NTTData
- Copiando el modelo de negocio de Oracle en cierto modo
 - Posicionamiento

Modelo de Negocio



Resumen: Ventajas

- **Orientación a Documentos.** La capacidad de almacenar directamente documentos en formato JSON convierte a MongoDB en una base de datos muy flexible y que reduce el tiempo de implementación de nuestras aplicaciones.
- MongoDB usa **expresiones regulares** para hacer las consultas, que son más potentes y flexibles que el lenguaje SQL. Además, existen numerosos drivers para lenguajes de programación como PyMongo para Python, Jongo para Java, mgo para Go o Mongoose para Node.js.
- **Alta disponibilidad.** MongoDB puede implementar replicación de los datos, manteniendo copias en varios nodos, que permiten mantener una alta disponibilidad y tolerancia a errores.
- **Escalabilidad.** Al repartir los datos en múltiples nodos con el mecanismo de sharding en replica sets, el sistema puede escalar horizontalmente y balancear la carga en varios servidores. Mediante una capa intermedia que actúa de proxy (Mongos) se dividen los datos en entornos de réplica. Se puede particionar mediante hash de manera equitativa o por campos.

Resumen: Recursos formativos

- MongoDB University → <https://learn.mongodb.com/> → “Certificación”
- Básico: Udemy
- Básico: Udemy
- Desarrollo Web Web: Udemy
- Recurso corto: Udemy
- Express JS: Recurso

Ejercicio Práctico

- Mongoshell Windows
- Cadena de Conexión → Instancia Atlas ó Clever Cloud

Ejercicio Extra

- Github → <https://github.com/IRasmivan/spring-boot-mongodb>
- Proyecto Full Stack - Back y Front
- Usar otros lenguajes y soluciones
- Para Matrícula: <https://github.com/artemadams/atlas-search-mongoflix>

GRACIAS