

REPORTE DE PRÁCTICA

NOMBRE DE LA PRÁCTICA: Consultas a la BD de Distribuidora
ALUMNO:

Gómez Daniel Aimar Jair



1. Introducción

En este reporte, se documenta la implementación de la base de datos mediante MongoDB en la distribuidora de herramientas. Durante la practica se realizo la instalación de MongoDB y la configuración de este, la creación de colecciones que se tuvieron que adaptar ya que venian de un modelo entidad-relación.

2. Marco teórico

1. Grado de la Relación El grado de la relación hace referencia al número de atributos (columnas) en una relación (tabla). En el contexto de bases de datos, una relación es una tabla y cada atributo es una columna en esa tabla. Por ejemplo, si una tabla "Empleado" tiene tres columnas (ID, nombre, salario), el grado de esa relación es 3.

2. Clave Candidata Una clave candidata es un conjunto de uno o más atributos de una tabla que puede identificar de manera única una fila en esa tabla. Una tabla puede tener varias claves candidatas, pero una de ellas se seleccionará como la clave primaria. Por ejemplo, en una tabla "Estudiante", tanto el número de matrícula como el correo electrónico pueden ser claves candidatas, ya que ambos pueden identificar de manera única a un estudiante.

3. Clave Primaria La clave primaria es una clave candidata seleccionada para identificar de manera única cada registro en una tabla. Una clave primaria no puede contener valores nulos y debe ser única en la tabla. Es esencial para mantener la integridad de los datos. Por ejemplo, en una tabla "Empleado", el ID del empleado podría ser la clave primaria, ya que cada empleado tiene un ID único.

4. Superclave Una superclave es un conjunto de atributos que puede identificar de manera única una fila en una tabla, pero a diferencia de la clave primaria, una superclave puede contener atributos adicionales que no son necesarios para la identificación única. Por ejemplo, en una tabla "Empleado", el conjunto de atributos ID, nombre puede ser una superclave, ya que, aunque el ID por sí solo puede identificar al empleado, agregar el nombre sigue siendo suficiente para identificarlo de manera única.

5. Clave Foránea Una clave foránea es un atributo (o conjunto de atributos) en una tabla que se utiliza para establecer una relación con la clave primaria de otra tabla. La clave foránea es crucial para la integridad referencial en las bases de datos. Por ejemplo, en una tabla "Pedido", el "ID_{cliente}" *puede ser una clave foránea que hace referencia a*

6. Cardinalidad La cardinalidad se refiere al número de elementos (tuplas o filas) en una relación o tabla. También describe las relaciones entre tablas en términos de los posibles valores que pueden tomar los atributos en una base de datos relacional. En el contexto de relaciones entre tablas, la cardinalidad también puede hacer referencia al número de instancias de una entidad que pueden estar asociadas a instancias de otra entidad (por ejemplo, uno a muchos, muchos a muchos).

7. Fragmentación Horizontal La fragmentación horizontal es una técnica de distribución de bases de datos en la que las filas (tuplas) de una tabla se distribuyen entre varias tablas basadas en un criterio de partición. Cada fragmento contiene un subconjunto de las filas originales de la tabla. Por ejemplo, en una tabla de "Ventas", se pueden crear fragmentos horizontales donde un fragmento contiene las ventas realizadas en un año y otro fragmento contiene las ventas de otro año.

8. Fragmentación Vertical La fragmentación vertical es un proceso mediante el cual las columnas de una tabla se distribuyen entre diferentes tablas. Este tipo de fragmentación es útil cuando se quiere reducir la cantidad de datos leídos por una consulta, dividiendo una tabla en fragmentos con diferentes columnas. Por ejemplo, una tabla de "Empleado" con columnas como "ID", "nombre", "salario" y "dirección" puede ser fragmentada verticalmente en dos tablas: una que contiene solo "ID" y "nombre" y otra que contiene "salario" y "dirección".

9. Índice Un índice es una estructura de datos que mejora la velocidad de las operaciones de búsqueda en una base de datos. Funciona como un índice en un libro, permitiendo localizar rápidamente los registros que cumplen con ciertas condiciones. Los índices se crean en columnas que se utilizan con frecuencia en las consultas, lo que mejora la eficiencia de las búsquedas, pero puede afectar el rendimiento de las operaciones de inserción y actualización.

10. Create View El comando CREATE VIEW en SQL se utiliza para crear una vista, que es una tabla

virtual basada en el resultado de una consulta SQL. Las vistas no almacenan datos por sí mismas, sino que almacenan una consulta que se ejecuta cada vez que la vista es accedida. Esto es útil para simplificar consultas complejas, restringir el acceso a ciertos datos y proporcionar una capa de abstracción sobre la estructura de la base de datos. Por ejemplo, una vista podría mostrar solo la información de los empleados que han trabajado más de 5 años, sin necesidad de modificar la estructura original de la tabla.

Este marco teórico cubre los conceptos clave que permiten entender cómo funcionan las bases de datos relacionales y cómo las relaciones, claves, fragmentaciones, índices y vistas ayudan en la organización y optimización de la gestión de datos.

3. Herramientas empleadas

1. MongoDB: Es una base de datos NoSQL orientada a documentos que permite almacenar datos de forma flexible, escalable y eficiente. En lugar de utilizar tablas y filas como en las bases de datos relacionales, MongoDB emplea documentos en formato JSON (JavaScript Object Notation) para representar los datos.

Desarrollo de los ejercicios

```
db.buytool.insertOne({
  idPurchase: ObjectId("6729985c75f62dc99cb4b142"),
  idInventory: ObjectId("6729a9b934a63e84021cd1a5"),
  amount: 92,
  unitPrice: "75.31",
  note: "Compra número 2"
})
```

```
db.inventory.insertOne({
  idWarehouse: ObjectId("67299aca75f62dc99cb4b14d"),
  idTool: ObjectId("67299a6775f62dc99cb4b145"),
  zone: "C",
  rack: 3,
  level: 2,
  location: "DD",
  amount: 75,
  purchasePrice: 31,
  storeSalePrice: 36,
  wholeSalePrice: 33
})
```

```
db.purchase.insertOne({
  idSupplier: ObjectId("6729949c978dbdfbe5c810da"),
  date: new Date("2024-01-10T00:00:00.000+00:00"),
  time: "10:30",
  folio: ObjectId("6729a9b934a63e84021cd19b"),
  payment: "Tarjeta de crédito"
})
```

```
db.supplier.insertOne({
  name: "Proveedor Uno",
  street: "Calle Principal",
  number: "123",
  city: "Ciudad Uno",
  state: "Estado Uno",
  phone: "555-1111",
  email: "contacto@proveedoruno.com",
  contact: "Juan Pérez"
})
```

```
db.tool.insertOne({
  name: "Destornillador",
  model: "DST-100",
  manufacturer: "Herramientas confiables",
  color: "Amarillo",
  size: "Mediano",
  description: "Destornillador multifuncional con cabeza intercambiable"
})
```

```

db.warehouse.insertOne({
  street: "Avenida Industrial",
  number: "200",
  city: "Ciudad Dos",
  state: "Estado Dos",
  area: "1500 m2",
  phone: "555-2222",
  manager: "Ana López"
})

```

Ejercicios consultas

1

```

db.getCollection('purchase').aggregate(
[
  {
    $lookup: {
      as: 'detallesProveedor',
      from: 'supplier',
      foreignField: '_id',
      localField: 'idSupplier'
    }
  },
  {
    $unwind: {
      path: '$detallesProveedor',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $match: {
      date: {
        $gte: ISODate(
          '2024-01-01T00:00:00.000Z'
        ),
        $lt: ISODate('2024-02-01T00:00:00.000Z')
      }
    }
  },
  {
    $project: {
      _id: 1,
      date: 1,
      'detallesProveedor.name': 1,
      folio: 1,
      idSupplier: 1,
      payment: 1,
      string: 1,
      time: 1
    }
  }
]
)

```

```

    }
  },
  {
    $lookup: {
      as: 'inventory',
      from: 'inventory',
      foreignField: '_id',
      localField: 'folio'
    }
  },
  {
    $lookup: {
      as: 'tool',
      from: 'tool',
      foreignField: '_id',
      localField: 'inventory.idTool'
    }
  },
  {
    $unwind: {
      path: '$inventory',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $unwind: {
      path: '$tool',
      includeArrayIndex: 'index',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $project: {
      _id: 1,
      date: 1,
      'detallesProveedor.name': 1,
      'tool.name': 1,
      'inventory.amount': 1,
      'inventory.storeSalePrice': 1,
      'inventory.wholeSalePrice': 1,
      payment: 1
    }
  }
],
{ maxTimeMS: 60000, allowDiskUse: true }
);

```

2

```

db.getCollection('inventory').aggregate(
[
  {

```



```

    $lookup: {
      as: 'warehouse',
      from: 'warehouse',
      foreignField: '_id',
      localField: 'idWarehouse'
    }
  },
  {
    $unwind: {
      path: '$warehouse',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $match: { 'warehouse.street': 'Calle 13' }
  },
  {
    $lookup: {
      as: 'tool',
      from: 'tool',
      foreignField: '_id',
      localField: 'idTool'
    }
  },
  {
    $unwind: {
      path: '$tool',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $project: {
      _id: 1,
      amount: 1,
      level: 1,
      location: 1,
      purchasePrice: 1,
      rack: 1,
      storeSalePrice: 1,
      'tool.name': 1,
      'warehouse.street': 1,
      zone: 1,
      costoTotal: {
        $multiply: [
          '$storeSalePrice',
          '$amount'
        ]
      }
    }
  }
],
{ maxTimeMS: 60000, allowDiskUse: true }

```

```
);
```

3

```
db.getCollection('purchase').aggregate([
  {
    $lookup: {
      as: 'inventory',
      from: 'inventory',
      foreignField: '_id',
      localField: 'folio'
    }
  },
  {
    $unwind: {
      path: '$inventory',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $lookup: {
      as: 'tool',
      from: 'tool',
      foreignField: '_id',
      localField: 'inventory.idTool'
    }
  },
  {
    $unwind: {
      path: '$tool',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $lookup: {
      as: 'supplier',
      from: 'supplier',
      foreignField: '_id',
      localField: 'idSupplier'
    }
  },
  {
    $unwind: {
      path: '$supplier',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $match: {
```

```

        date: {
          $gte: ISODate(
            '2023-01-01T00:00:00.000Z'
          ),
          $lt: ISODate('2024-01-01T00:00:00.000Z')
        }
      },
      {
        $match: {
          'tool.name': 'Tijera de jardinero'
        }
      },
      {
        $project: {
          date: 1,
          _id: 1,
          time: 1,
          'tool.name': 1,
          'inventory.amount': 1,
          'inventory.storeSalePrice': 1,
          'inventory.wholeSalePrice': 1,
          'inventory.purchasePrice': 1,
          payment: 1,
          costoTotal: {
            $multiply: [
              '$inventory.storeSalePrice',
              '$inventory.amount'
            ]
          }
        }
      }
    ],
    { maxTimeMS: 60000, allowDiskUse: true }
  );

```

4

```

db.getCollection('warehouse').aggregate(
  [
    {
      $project: {
        manager: 1,
        street: 1,
        number: 1,
        phone: 1
      }
    }
  ],
  { maxTimeMS: 60000, allowDiskUse: true }
);

```

5

```
db.getCollection('supplier').aggregate([
  {
    $project: {
      contact: 1,
      name: 1,
      phone: 1,
      email: 1
    }
  },
  { maxTimeMS: 60000, allowDiskUse: true }
]);
```

6

```
db.getCollection('buytool').aggregate([
  {
    $lookup: {
      as: 'purchase',
      from: 'purchase',
      foreignField: '_id',
      localField: 'idPurchase'
    }
  },
  {
    $unwind: {
      path: '$purchase',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $lookup: {
      as: 'inventory',
      from: 'inventory',
      foreignField: '_id',
      localField: 'idInventory'
    }
  },
  {
    $unwind: {
      path: '$inventory',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $lookup: {
      as: 'tool',
```

```

        from: 'tool',
        foreignField: '_id',
        localField: 'inventory.idTool'
    }
},
{
    $unwind: {
        path: '$tool',
        includeArrayIndex: 'string',
        preserveNullAndEmptyArrays: true
    }
},
{
    $project: {
        'tool.name': 1,
        'purchase.date': 1,
        amount: 1,
        unitPrice: 1
    }
},
{ $match: { unitPrice: { $lte: '250.00' } } },
{ $sort: { 'purchase.date': -1 } }
],
{ maxTimeMS: 60000, allowDiskUse: true }
);

```

7

```

db.getCollection('inventory').aggregate(
[
    {
        $lookup: {
            as: 'warehouse',
            from: 'warehouse',
            foreignField: '_id',
            localField: 'idWarehouse'
        }
    },
    {
        $unwind: {
            path: '$warehouse',
            includeArrayIndex: 'string',
            preserveNullAndEmptyArrays: true
        }
    },
    {
        $project: {
            'warehouse.street': 1,
        }
    }
],
{ maxTimeMS: 60000, allowDiskUse: true }
);

```

8

```
db.getCollection('inventory').aggregate([
  {
    $lookup: {
      as: 'warehouse',
      from: 'warehouse',
      foreignField: '_id',
      localField: 'idWarehouse'
    }
  },
  {
    $unwind: {
      path: '$warehouse',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $lookup: {
      as: 'tool',
      from: 'tool',
      foreignField: '_id',
      localField: 'idTool'
    }
  },
  {
    $unwind: {
      path: '$tool',
      includeArrayIndex: 'string',
      preserveNullAndEmptyArrays: true
    }
  },
  {
    $addFields: {
      montoTotal: {
        $multiply: [
          '$storeSalePrice',
          '$amount'
        ]
      }
    }
  },
  {
    $project: {
      'warehouse.street': 1,
      'warehouse.state': 1,
      montoTotal: 1
    }
  }
],
{ maxTimeMS: 60000, allowDiskUse: true }
);
```

1

<pre> _id: ObjectId('6729985c75f62dc99cb4b13a') date: 2024-01-10T00:00:00.000+00:00 payment: "Tarjeta de crédito" ▾ detallesProveedor: Object name: "Proveedor Uno" ▾ inventory: Object amount: 50 storeSalePrice: 15 wholeSalePrice: 12 ▾ tool: Object name: "martillo" </pre>
<pre> _id: ObjectId('6729985c75f62dc99cb4b13b') date: 2024-01-15T00:00:00.000+00:00 payment: "Efectivo" ▾ detallesProveedor: Object name: "Proveedor Dos" ▾ inventory: Object amount: 75 storeSalePrice: 25 wholeSalePrice: 22 ▾ tool: Object name: "Destornillador" </pre>
<pre> _id: ObjectId('6729985c75f62dc99cb4b13e') date: 2024-01-30T00:00:00.000+00:00 payment: "Tarjeta de crédito" ▾ detallesProveedor: Object name: "Proveedor Dos" ▾ inventory: Object amount: 100 storeSalePrice: 35 wholeSalePrice: 32 ▾ tool: Object name: "Taladro" </pre>

2

<div><div></div><div><pre> _id: ObjectId('6729a9b934a63e84021cd19d') zone: "C" rack: 3 level: 3 location: "DD" amount: 100 purchasePrice: 30 storeSalePrice: 35 ▾ warehouse: Object street: "Calle 13" ▾ tool: Object name: "Taladro" costoTotal: 3500 </pre></div><div></div></div>
<div><div></div><div><pre> _id: ObjectId('6729a9b934a63e84021cd1a1') zone: "B" rack: 2 level: 3 location: "SC" amount: 65 purchasePrice: 22 storeSalePrice: 28 ▾ warehouse: Object street: "Calle 13" ▾ tool: Object name: "Destornillador" costoTotal: 1820 </pre></div><div></div></div>
<div><div></div><div><pre> _id: ObjectId('6729a9b934a63e84021cd1a4') zone: "E" rack: 5 level: 2 location: "GC" amount: 70 purchasePrice: 27 storeSalePrice: 33 ▾ warehouse: Object street: "Calle 13" </pre></div><div></div></div>

3

```

_id: ObjectId('6729aab58d83f803c95837ad')
date: 2023-08-28T00:00:00.000+00:00
time: "16:30"
payment: "Efectivo"
inventory: Object
  amount: 30
  purchasePrice: 13
  storeSalePrice: 20
  wholeSalePrice: 18
tool: Object
  name: "Tijera de jardinero"
costoTotal: 600
```

4

<pre> _id: ObjectId('67299aca75f62dc99cb4b149') street: "Calle Central" number: "100" phone: "555-1111" manager: "Juan Pérez"</pre>
<pre> _id: ObjectId('67299aca75f62dc99cb4b14a') street: "Avenida Industrial" number: "200" phone: "555-2222" manager: "Ana López"</pre>
<pre> _id: ObjectId('67299aca75f62dc99cb4b14b') street: "Boulevard Comercial" number: "300" phone: "555-3333" manager: "Carlos García"</pre>
<pre> _id: ObjectId('67299aca75f62dc99cb4b14c') street: "Calle 13" number: "400" phone: "555-4444" manager: "María Hernández"</pre>
<pre> _id: ObjectId('67299aca75f62dc99cb4b14d') street: "Avenida del Sur" number: "500" phone: "555-5555" manager: "Luis Martínez"</pre>


```
_id: ObjectId('6729949c978dbdfbe5c810da')  
name : "Proveedor Uno"  
phone : "555-1111"  
email : "contacto@proveedoruno.com"  
contact : "Juan Pérez"
```

▶

```
_id: ObjectId('6729949c978dbdfbe5c810db')  
name : "Proveedor Dos"  
phone : "555-2222"  
email : "contacto@proveedordos.com"  
contact : "Ana López"
```

```
_id: ObjectId('6729949c978dbdfbe5c810dc')  
name : "Proveedor Tres"  
phone : "555-3333"  
email : "contacto@proveedortres.com"  
contact : "Carlos García"
```

```
_id: ObjectId('6729949c978dbdfbe5c810dd')  
name : "Proveedor Cuatro"  
phone : "555-4444"  
email : "contacto@proveedorcuatro.com"  
contact : "María Hernández"
```

```
_id: ObjectId('6729949c978dbdfbe5c810de')  
name : "Proveedor Cinco"  
phone : "555-5555"  
email : "contacto@proveedorcinco.com"  
contact : "Luis Martínez"
```

6

```
_id: ObjectId('6729ae553e4c5aacfe6d314b')
amount: 56
unitPrice: "1.76"
▼ purchase: Object
  date: 2024-06-24T00:00:00.000+00:00
▼ tool: Object
  name: "Destornillador"
```

```
_id: ObjectId('6729ae553e4c5aacfe6d3145')
amount: 24
unitPrice: "0.91"
▼ purchase: Object
  date: 2024-05-22T00:00:00.000+00:00
▼ tool: Object
  name: "Taladro"
```

```
_id: ObjectId('6729ae553e4c5aacfe6d314a')
amount: 41
unitPrice: "11.19"
▼ purchase: Object
  date: 2024-03-05T00:00:00.000+00:00
▼ tool: Object
  name: "Tijera de jardinero"
```

```
_id: ObjectId('6729ae553e4c5aacfe6d313e')
amount: 64
unitPrice: "0.51"
▼ purchase: Object
  date: 2024-02-25T00:00:00.000+00:00
▼ tool: Object
  name: "Taladro"
```

7



```
_id: ObjectId('6729a9b934a63e84021cd1a0')
zone : "A"
amount : 7
▼ warehouse : Object
  street : "Calle Central"
  state : "Estado Uno"
  phone : "555-1111"
▼ tool : Object
  name : "martillo"
```

```
_id: ObjectId('6729a9b934a63e84021cd1a6')
zone : "B"
amount : 20
▼ warehouse : Object
  street : "Calle Central"
  state : "Estado Uno"
  phone : "555-1111"
```

```
_id: ObjectId('6729a9b934a63e84021cd1a8')
zone : "D"
amount : 19
▼ warehouse : Object
  street : "Boulevard Comercial"
  state : "Estado Tres"
  phone : "555-3333"
▼ tool : Object
  name : "Taladro"
```

```
_id: ObjectId('6729a9b934a63e84021cd1ac')
zone : "C"
amount : 6
▼ warehouse : Object
  street : "Avenida del Sur"
  state : "Estado Cinco"
  phone : "555-5555"
▼ tool : Object
  name : "Destornillador"
```

8

```
_id: ObjectId('6729a9b934a63e84021cd19b')
▼ warehouse : Object
  street : "Calle Central"
  state : "Estado Uno"
montoTotal : 750
```

```
_id: ObjectId('6729a9b934a63e84021cd19c')
▼ warehouse : Object
  street : "Avenida Industrial"
  state : "Estado Dos"
montoTotal : 1875
```

```
_id: ObjectId('6729a9b934a63e84021cd19d')
▼ warehouse : Object
  street : "Calle 13"
  state : "Estado Cuatro"
montoTotal : 3500
```

```
_id: ObjectId('6729a9b934a63e84021cd19e')
▼ warehouse : Object
  street : "Avenida del Sur"
  state : "Estado Cinco"
montoTotal : 1200
```

```
_id: ObjectId('6729a9b934a63e84021cd19f')
▼ warehouse : Object
  street : "Avenida del Sur"
  state : "Estado Cinco"
montoTotal : 2400
```

```
_id: ObjectId('6729a9b934a63e84021cd1a0')
▼ warehouse : Object
  street : "Calle Central"
  state : "Estado Uno"
montoTotal : 126
```

Álgebra relacional

$$\pi_{\text{supplierName,toolName,amount,unitPrice,(amount*unitPrice)} \text{ AS TotalPrice}} \\ (\sigma(\text{MONTH}(\text{date}) = 1) ((\text{purchase} \bowtie \text{supplier}) \bowtie (\text{buyTool} \bowtie \text{inventory} \bowtie \text{tool}))) \quad (1)$$

$$\pi_{\text{toolName,amount,(purchasePrice*amount)} \text{ AS TotalCost}} \\ (\sigma(\text{street} = '13') ((\text{inventory} \bowtie \text{warehouse} \bowtie \text{tool}))) \quad (2)$$

$$\pi_{\text{supplierName,date,amount,unitPrice,(amount*unitPrice)} \text{ AS TotalCost}} \\ (\sigma(\text{name} = 'Tijeradejardinero' \wedge \text{YEAR}(\text{date}) = 2023) ((\text{purchase} \bowtie \text{supplier}) \bowtie (\text{buyTool} \bowtie \text{inventory} \bowtie \text{tool}))) \quad (3)$$

$$\pi_{\text{street,number,phone,manager}} (\text{warehouse})$$

$$\pi_{\text{contact,supplierName,phone,email}} (\text{supplier})$$

$$\pi_{\text{toolName,date,amount}} \\ (\sigma(\text{unitPrice} \leq 250) ((\text{purchase} \bowtie \text{buyTool} \bowtie \text{inventory} \bowtie \text{tool}))) \quad (4)$$

$$\pi_{\text{street,number,toolName,zone,rack,level,location,amount}} \\ (\sigma(\text{amount} \geq 5 \wedge \text{amount} \leq 20) ((\text{inventory} \bowtie \text{warehouse} \bowtie \text{tool}))) \quad (5)$$

$$\pi_{\text{street,number,manager,phone,SUM(amount)} \text{ AS TotalTools}} \\ ((\text{warehouse} \bowtie \text{inventory})) \quad (6)$$

Conclusiones

En mi opinion el uso de MongoDB permitió manejar los datos de manera más ágil en comparación con una base de datos relacional tradicional, especialmente debido a su naturaleza flexible y la posibilidad de trabajar con documentos JSON que pueden adaptarse fácilmente a cambios futuros

Referencias Bibliográficas

References

- [1] DATAPRIX. (2023, enero 8). Clave candidata, clave primaria y clave alternativa de las relaciones. *DATAPRIX*. Recuperado de <https://www.dataprix.com/es/bases-datos-master-software-libre-uoc/24-clave-candidata-clave-primaria-y-clave-alternativa>.
- [2] Informaticos sin límites. (2022). Superclaves en base de datos + Ejemplos. *Informaticos sin límites*. Recuperado de <https://informaticosinlimites.com/base-de-datos/superclaves/>.
- [3] Ravooof, S. (2023, febrero 24). Una guía para entender la fragmentación de las bases de datos. *Kinsta*. Recuperado de <https://kinsta.com/es/blog/fragmentacion-de-la-base-de-datos/:.text=La20fragmentaciC3B3n20horizontal20divide20la,grandes20en20otras20mC3A1s20pequeC3B1as>.