



**Universidad Central de Venezuela
Facultad de Ciencias Escuela
de Computación
Licenciatura en Computación**



Base de Datos NoSQL

Proyecto Fase 1

Abelardo Moreno C.I: 18.002.106

David Fernandez C.I: 24.313.648

Alberto Suarez C.I: 23.107.093

- **Modelo de datos:**

El modelo de datos utilizado fue un modelo embebido, en el que en una sola colección se almacenaron todos los documentos con la información de la criaturas. Esto fue porque no había realmente información que pudiera ser almacenada en otra colección y referenciada en nuestra colección de criaturas.

Debido a que el modelo es completamente embebido, la inserción y modificación de un campo es un proceso bastante engorroso pero las consultas se realizan con mucha facilidad.

- **Script:**

Para el script, utilizamos una variable donde se almacenan todos los documentos que vamos a almacenar para luego ser insertados en la BD. Creamos la BD que vamos a utilizar "ProyectoCriaturas" y la colección "criaturas" con un índice en el campo _id.

Después procedemos a la creación de un índice sobre el campo "clasificación" como se indica en el enunciado, para hacer más rápidas las búsquedas por este campo.

Luego procedemos a la creación de los usuarios "Ministro", "Scamander" y "Mago" los cuales se les asignan los privilegios indicados en el enunciado.

- **Explain:**

Para obtener el explain plan de una consulta en mongo, se debe de usar la función .explain("executionStats") la cual nos provee de cierta información que nos permitirá analizar información tal como si se estan y como se están utilizando los índices en la consulta.

En nuestro caso realizamos el explain sobre la siguiente consulta:
`db.criaturas.find({"clasificacion":"spirit"}).explain("executionStats");`

Y obtuvimos los siguientes resultados

```

{ "_id" : "reptil", "promedio_tiempo_de_vida" : 5400.0005 }
{
  "queryPlanner" : {
    "plannerVersion" : 1,
    "namespace" : "ProyectoCriaturas.criaturas",
    "indexFilterSet" : false,
    "parsedQuery" : {
      "clasificacion" : {
        "$eq" : "spirit"
      }
    }
  },
  "winningPlan" : {
    "stage" : "FETCH",
    "inputStage" : {
      "stage" : "IXSCAN",
      "keyPattern" : {
        "clasificacion" : 1
      },
      "indexName" : "clasificacion_1",
      "isMultiKey" : false,
      "multiKeyPaths" : {
        "clasificacion" : [ ]
      },
      "isUnique" : false,
      "isSparse" : false,
      "isPartial" : false,
      "indexVersion" : 2,
      "direction" : "forward",
      "indexBounds" : {
        "clasificacion" : [
          ["\"spirit\"", "\"spirit\""]
        ]
      }
    }
  },
  "rejectedPlans" : [ ]
}

```

El campo winningPlan.inputState.stage; "IXSCAN" nos indica que la consulta está utilizando un índice.

```

"executionStats" : {
  "executionSuccess" : true,
  "nReturned" : 7,
  "executionTimeMillis" : 0,
  "totalKeysExamined" : 7,
  "totalDocsExamined" : 7,
  "executionStages" : {
    "stage" : "FETCH",
    "nReturned" : 7,
    "executionTimeMillisEstimate" : 0,
    "works" : 8,
    "advanced" : 7,
    "needTime" : 0,
    "needYield" : 0,
    "saveState" : 0,
    "restoreState" : 0,
    "isEOF" : 1,
    "invalidates" : 0,
    "docsExamined" : 7,
    "alreadyHasObj" : 0,
    "inputStage" : {
      "stage" : "IXSCAN",
      "nReturned" : 7,
      "executionTimeMillisEstimate" : 0,
      "works" : 8,
      "advanced" : 7,
      "needTime" : 0,
      "needYield" : 0,
      "saveState" : 0,
      "restoreState" : 0,
      "isEOF" : 1,
      "invalidates" : 0,
      "keyPattern" : {
        "clasificacion" : 1
      },
      "indexName" : "clasificacion_1",
      "isMultiKey" : false,
      "multiKeyPaths" : {
        "clasificacion" : [ ]
      }
    }
  }
}

```

El campo executionStats.nReturned nos indica 7, que significa que la consulta hace match y retorna 3 documentos.

El campo executionStats.totalKeysExamined nos indica 7 que significa que Mongo escaneo 7 keys con indice.

El campo executionStats.totalDocsExamined nos indica 7 que indica que Mongo escaneo 7 documentos

Con esto podemos ver que la consulta con índices es muy eficiente ya que no tiene recorrer todos los documentos en la BD para obtener los que se están buscando en la consulta.