

Information Management & Systems Engineering

Milestone 2: Migration from a relational DBMS to NoSQL

11807201 Frauenberger Fabian

11911464 Mitterrutzner Samuel

A dark blue diagonal gradient bar that starts from the bottom left and extends towards the top right, covering the lower half of the slide.

Architecture and Technology Stack

Frontend:

- NextJS

Backend:

- NodeJS
- ExpressJS (REST API)
- Sequelize ORM
- Mongoose

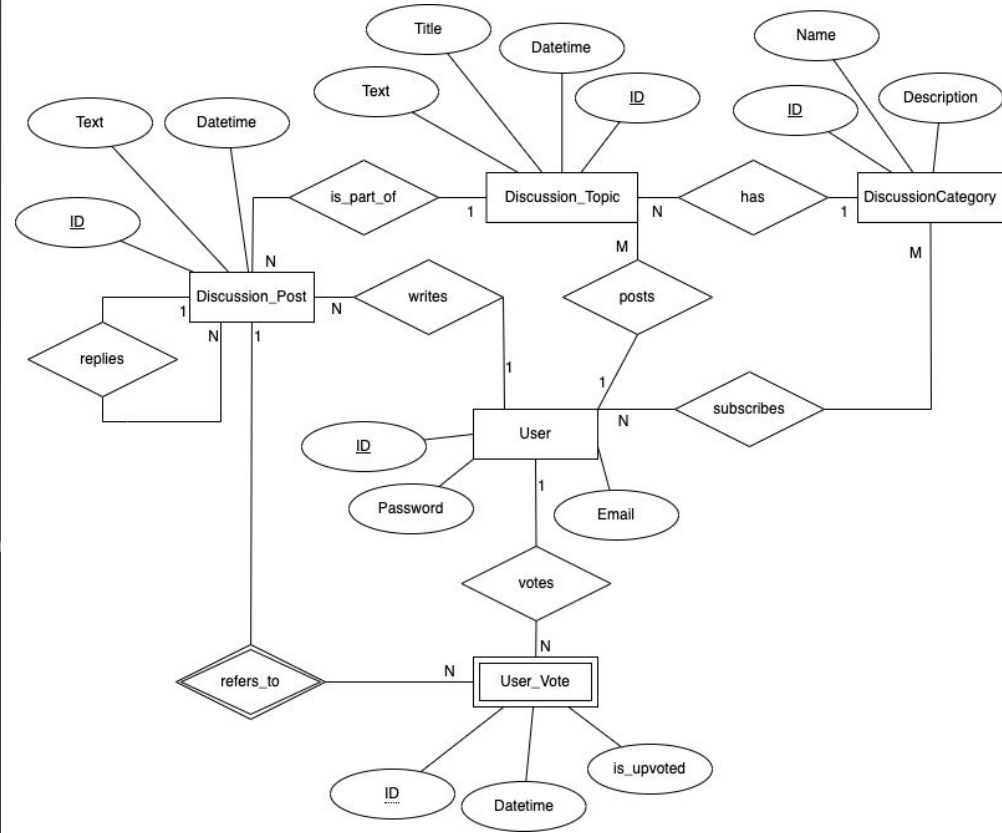
Database:

- mysql 8.0
- adminer (for web-access)
- mongoDB
- mongo express (for web-access)

Web server:

- NGINX

an online platform where people can discuss
anonymously



Comparison RDBMS – NoSQL

Discussion Category

discussionCategories (id, name, description)

PK: name

```
{
  _id: ObjectId('62b2c6f1abc30fb4b02db9f1'),
  description: 'Local News description',
  name: 'Local News',
}
```

Comparison RDBMS – NoSQL

Users

users (id,email,password)

PK: id

subscriptions (DiscussionCategoryId, UserId)

PK: DiscussionCategoryId, UserId

FK: subscriptions.DiscussionCategoryId <> discussionCategories.id

subscriptions.UserId <> users.id

```
{
  _id: ObjectId('62b2c6f1abc30fb4b02db9e7'),
  email: 'Levi_Gerhold@yahoo.com',
  password: 'RtrKFGapBGzI7dN',
  subscriptions: [
    ObjectId('62b2c6f1abc30fb4b02db9f1'),
    ObjectId('62b2c6f1abc30fb4b02db9f3'),
    ObjectId('62b2c6f1abc30fb4b02db9f5'),
    ObjectId('62b2c6f1abc30fb4b02db9f8')
  ],
}
```

Comparison RDBMS – NoSQL

Discussion Topic

discussion_topics (id, title, text, datetime, *discussionCategoryId*, *userId*)

PK: id

FK: discussion_topics.discussionCategoryId <> discussionCategories.id

discussion_topics.userId <> users.id

```
{
  _id: ObjectId('62b2c6f1abc30fb4b02dba20'),
  datetime: ISODate('2021-12-23T07:45:59.000Z'),
  discussionCategory: ObjectId('62b2c6f1abc30fb4b02db9fb'),
  text: 'Ad quos sed nemo.\nDucimus odit fugiat.\nProvident quae ut omnis quod.',
  title: 'topic 24',
  userId: '62b2c6f1abc30fb4b02db9e7',
  postsCount: 6,
}
```

Comparison RDBMS – NoSQL

Discussion Post

discussionPosts (id, text, datetime, *discussionThemeld*, *userId*, *parentPostId*)
PK: id
FK: discussionPosts.discussionThemeld <> discussion_topics.id
discussionPosts.userId <> users.id
discussionPosts.parentPostId <> discussionPosts.id

userVotes (id, discussionPostId, isUpvote, datetime, *userId*)
PK: id, discussionPostId
FK: userVotes.discussionPostId <> discussionPosts.id
userVotes.userId <> users.id

```
_id: ObjectId('62b2c6f2abc30fb4b02dba48'),  
  
text: 'Nostrum facere ipsa molestiae molestiae eius veniam quasi.',  
  
datetime: ISODate('2022-05-11T11:57:21.000Z'),  
  
discussionThemeId: '62b2c6f1abc30fb4b02dba20',  
  
parentPostId: null,  
  
userId: '62b2c6f1abc30fb4b02db9e7',  
  
userVotes: [  
  
  {  
  
    datetime: ISODate('2022-05-20T00:27:21.000Z'),  
  
    isUpvote: true,  
  
    userId: '62b2c6f1abc30fb4b02db9eb',  
  
    _id: ObjectId('62b2c6f2abc30fb4b02dba49')  
  
  }  
  
],  
  
}
```

Comparison of SQL query to the MongoDB query

Creating Discussion Topic

```
INSERT INTO discussion_topics

(datetime,

id,

text,

title,

userid,

discussioncategoryid)

VALUES ('2021-06-22 20:25:00',

DEFAULT,

'Test text',

'test',

2,

1)
```

```
db.discussiontopics.insertOne({
  text: 'some text',
  title: 'title',
  datetime: '2021-06-22 20:25:00',
  userId: '62b2c6f1abc30fb4b02db9e7',
  discussionCategory: '62b2c6f1abc30fb4b02db9fb',
  postsCount: 0
})
```


Comparison of SQL query to the MongoDB query

Writing a Post

```
INSERT INTO discussionposts

    (text,

    datetime,

    id,

    discussionthemeid

    userid)

VALUES ('testing',

    '2021-06-22 20:25:00',

    DEFAULT,

    6,

    3)
```

```
db.discussionPosts.insertOne({
  text: 'Nostrum facere ipsa molestiae molestiae eius
veniam quasi.',
  datetime: ISODate('2022-05-11T11:57:21.000Z'),
  discussionThemeId: '62b2c6f1abc30fb4b02dba20',
  parentPostId: null,
  userId: '62b2c6f1abc30fb4b02db9e7',
  userVotes: []
})

db.discussionTopics.updateOne({
  _id: '62b2c6f1abc30fb4b02dba20'
}, {
  $inc: {
    postsCount: 1
  }
})
```

Comparison of SQL query to the MongoDB query

Which are interesting topics of discussions?

```
SELECT DiscussionTopic.datetime,
       DiscussionTopic.id,
       DiscussionTopic.text,
       DiscussionTopic.title,
       DiscussionTopic.userid,
       DiscussionTopic.discussioncategoryid,
       DiscussionTopic.createdat,
       DiscussionTopic.updatedat,
       Count(discussionposts.id) AS PostCount,
       DiscussionCategory.NAME AS discussioncategory.NAME,
       DiscussionCategory.description AS
discussioncategory.description,
       DiscussionCategory.id AS discussioncategory.id,
       DiscussionCategory.createdat AS
discussioncategory.createdAt,
       DiscussionCategory.updatedat AS
discussioncategory.updatedAt
FROM   discussion_topics AS DiscussionTopic
       LEFT OUTER JOIN discussioncategories AS DiscussionCategory
           ON DiscussionTopic.discussioncategoryid =
              DiscussionCategory.id
       LEFT OUTER JOIN discussionposts AS DiscussionPosts
           ON DiscussionTopic.id =
discussionposts.discussionthemeid
WHERE  DiscussionTopic.datetime >= '2021-06-22 20:25:00'
GROUP BY DiscussionTopic.id
ORDER BY postcount DESC
```

```
db.discussiontopics.aggregate([
  $match: {
    datetime: {
      $gte: ISODate('2021-12-23T07:45:59.000Z')
    }
  },
  {
    $sort: {
      postsCount: -1
    }
  },
  {
    $lookup: {
      from: 'discussioncategories',
      localField: 'discussionCategory',
      foreignField: '_id',
      as: 'discussionCategory'
    }
  },
  {
    $unwind: {
      path: '$discussionCategory'
    }
  }
])
```

Comparison of SQL query to the MongoDB query

What are all the topics a user has created sorted by category

```
SELECT DiscussionTopic.datetime,  
       DiscussionTopic.id,  
       DiscussionTopic.text,  
       DiscussionTopic.title,  
       USER.id,  
       DiscussionTopic.discussioncategoryid,  
       DiscussionTopic.createdat,  
       DiscussionTopic.updatedat,  
       DiscussionCategory.NAME           AS  
discussioncategory.NAME,  
       DiscussionCategory.description AS  
discussioncategory.description,  
       DiscussionCategory.id           AS discussioncategory.id,  
       DiscussionCategory.createdat    AS  
discussioncategory.createdat,  
       DiscussionCategory.updatedat    AS  
discussioncategory.updatedat  
FROM   discussion_topics AS DiscussionTopic  
       LEFT OUTER JOIN discussioncategories AS  
DiscussionCategory  
           ON DiscussionTopic.discussioncategoryid =  
              DiscussionCategory.id  
       LEFT OUTER JOIN users AS USER  
           ON DiscussionTopic.userid = USER.id  
WHERE  DiscussionTopic.userid = '1'  
ORDER BY DiscussionCategory.NAME ASC
```

```
db.discussiontopics.aggregate([[  
    $match: {  
        userId: '62b38509d6ff45edc0a3614d'  
    }  
], [  
    $lookup: {  
        from: 'discussioncategories',  
        localField: 'discussionCategory',  
        foreignField: '_id',  
        as: 'discussionCategory'  
    }  
], [  
    $unwind: {  
        path: '$discussionCategory'  
    }  
], [  
    $sort: {  
        'discussionCategory.name': 1  
    }  
]])
```

Live Demo

