

## ACIT 3695 Lab 5

Team ScrapQL  
Michael Chau A01073751  
Tristan Priestley A01089705

### MongoDb Creation

Step 1: Creating the project and adding a partner to the project

BCIT

#### Projects

New Project

Find a project...

Project Name	Clusters	Users	Teams	Alerts	Actions
3695-notes-api	1 Cluster	2 Users	0 Teams	0 Alerts	...

Step 2: Creating a cluster

BCIT > 3695-NOTES-API

#### Clusters

Create a New Cluster

Find a cluster...

SANDBOX

graphql-api

Version 4.2.10

CONNECT

METRICS

COLLECTIONS

...

CLUSTER TIER

MO Sandbox (General)

REGION

AWS / Oregon (us-west-2)


TYPE

Replica Set - 3 nodes

LINKED REALM APP


None Linked

Operations R: 0 W: 0



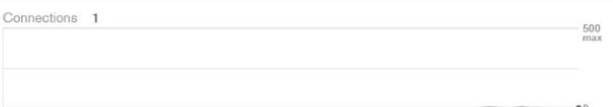
Last 6 Hours

Logical Size 0.0 B



Last 30 Days

Connections 1



Last 6 Hours

Enhance Your Experience

For dedicated throughput, richer metrics and enterprise security options, upgrade your cluster now!

Upgrade

### Step 3: Connecting to the database. Screenshot of index.js

```
JS index.js x JS resolvers.js JS typeDefs.js
src > JS index.js > ...
1 import apolloServer from 'apollo-server'
2 import mongoose from 'mongoose'
3 import { typeDefs } from './models/typeDefs.js'
4 import { resolvers } from './models/resolvers.js'
5
6 const { ApolloServer } = apolloServer
7
8
9 const start = async () => {
10
11   await mongoose.connect("mongodb+srv://[REDACTED]:[REDACTED]@graphql-api.1a5w5.mongodb.net/graphqlNotes?retryWrites=true&w=majority", {
12     useNewUrlParser: true,
13     useUnifiedTopology: true
14   })
15
16   const server = new ApolloServer({typeDefs, resolvers})
17   server.listen({port: 4000}).then(({url}) => console.log(`server running at ${url}`))
18
19 }
20
21 start()
22
```

### Screenshot of the database/notes

The screenshot shows the MongoDB Atlas web interface. The 'Collections' tab is selected, displaying the 'graphqlNotes.notes' collection. The collection has a size of 268B, 2 total documents, and 36KB of indexes. A filter bar is present with the text 'FILTER {"filter":"example"}'. Below the filter, the query results are shown for 1-2 of 2 documents.

**graphqlNotes.notes**  
COLLECTION SIZE: 268B TOTAL DOCUMENTS: 2 INDEXES TOTAL SIZE: 36KB

**Find** Indexes Schema Anti-Patterns Aggregation Search Indexes

**INSERT DOCUMENT**

**FILTER** {"filter":"example"} **Find** **Reset**

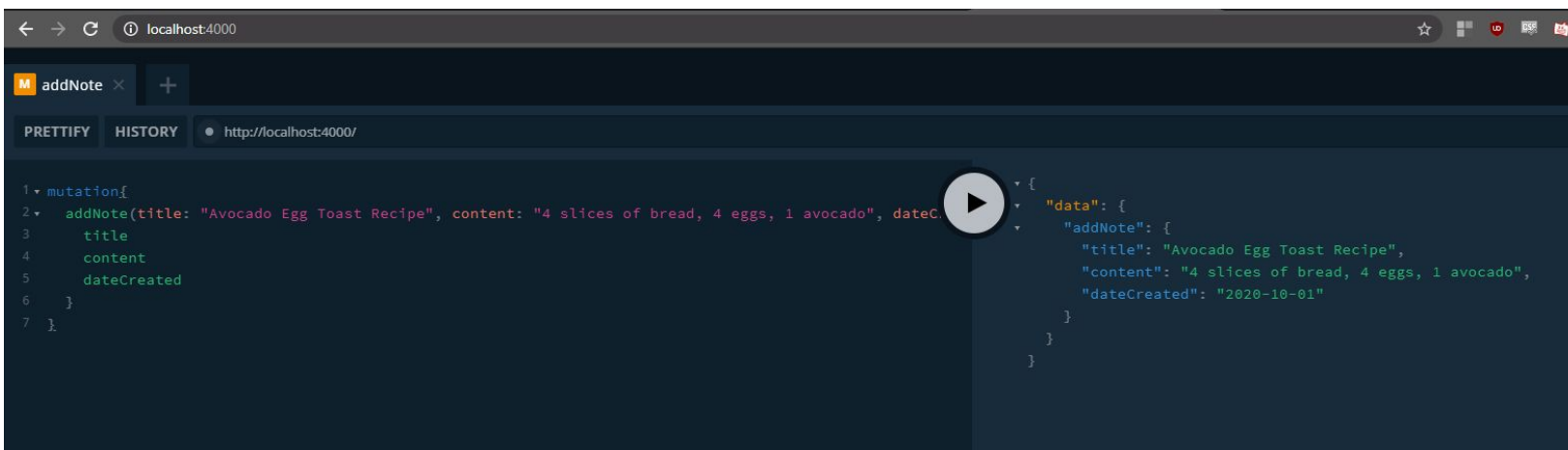
**QUERY RESULTS 1-2 OF 2**

```
{
  "_id": "ObjectId('5f7fc7df39ddfcfa8a800c1')",
  "title": "Avocado Egg Toast Recipe",
  "content": "4 slices of bread, 4 eggs, 1 avocado",
  "dateCreated": "2020-10-01",
  "__v": 0
}
```

```
{
  "_id": "ObjectId('5f7fca9b5c8774540469f5e')",
  "title": "How to become the best at GraphQL",
  "content": "Learn",
  "dateCreated": "2020-10-08",
  "__v": 0
}
```

Rest of the code: <https://github.com/MichaelPhuc/3695-graphql>

Mutation 1: adding a note:

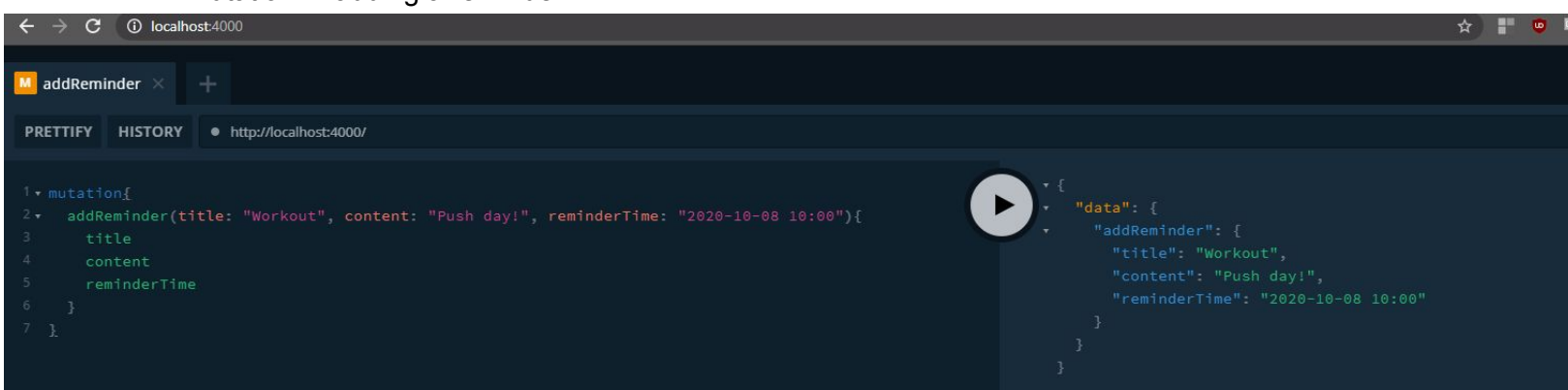


The screenshot shows a web browser at localhost:4000 with a tab titled 'addNote'. The browser's developer tools are open, displaying a GraphQL query and its JSON response. The query is a mutation to add a note, and the response shows the successful addition of the note with a timestamp.

```
1 mutation{
2   addNote(title: "Avocado Egg Toast Recipe", content: "4 slices of bread, 4 eggs, 1 avocado", dateCreated: "2020-10-01") {
3     title
4     content
5     dateCreated
6   }
7 }
```

```
{
  "data": {
    "addNote": {
      "title": "Avocado Egg Toast Recipe",
      "content": "4 slices of bread, 4 eggs, 1 avocado",
      "dateCreated": "2020-10-01"
    }
  }
}
```

Mutation 2: adding a reminder:

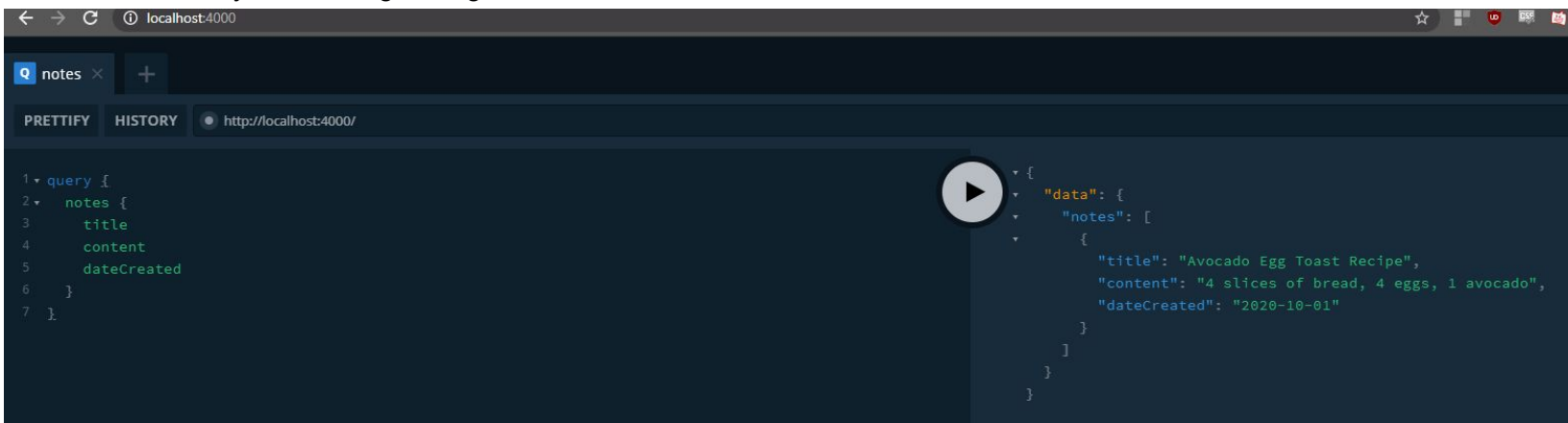


The screenshot shows a web browser at localhost:4000 with a tab titled 'addReminder'. The browser's developer tools are open, displaying a GraphQL query and its JSON response. The query is a mutation to add a reminder, and the response shows the successful addition of the reminder with a timestamp.

```
1 mutation{
2   addReminder(title: "Workout", content: "Push day!", reminderTime: "2020-10-08 10:00"){
3     title
4     content
5     reminderTime
6   }
7 }
```

```
{
  "data": {
    "addReminder": {
      "title": "Workout",
      "content": "Push day!",
      "reminderTime": "2020-10-08 10:00"
    }
  }
}
```

### Query 1: Showing a Single Note:



### Query 2: Showing a Single Reminder:

