

PROIECT WEB SEMANTIC

MEMBRII ECHIPĂ:

LORINCZ MARIUS-LAURENȚIU

PITIRICIU IULIAN

AN UNIVERSITAR 2019-2020



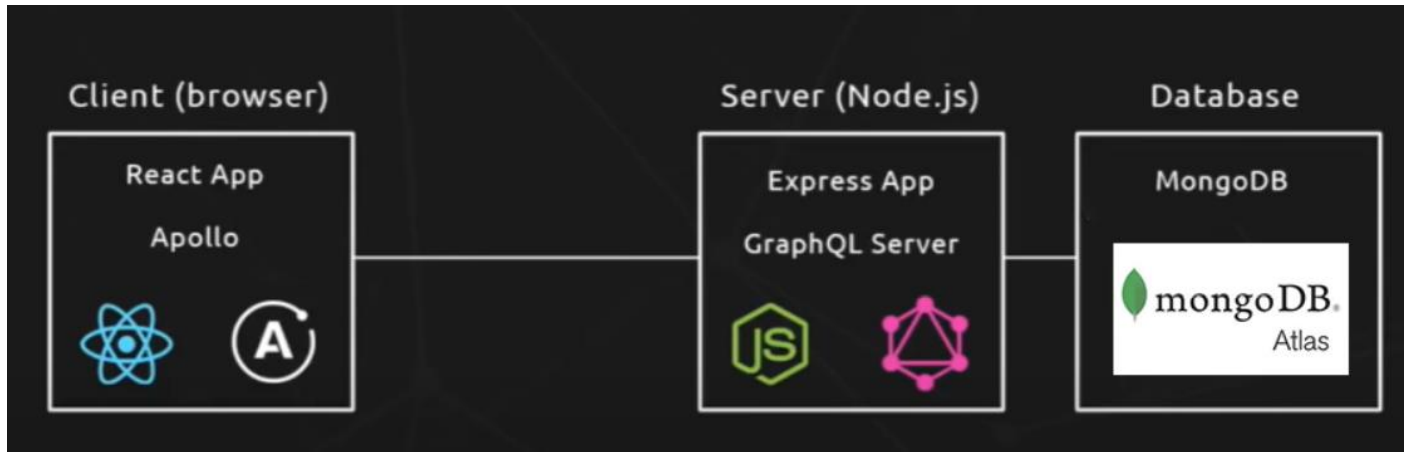
CUPRINS

| | |
|--|-----------|
| INTRODUCERE..... | 3 |
| TEHNOLOGIILE FOLOSITE..... | 4 |
| PACKAGE-URI FOLOSITE..... | 4 |
| SCREENSHOT-URI FACUTE ÎN TIMP CE DEZVOLTAM APLICAȚIA..... | 6 |
| BIBLIOGRAFIE..... | 11 |

INTRODUCERE

Aplicația noastră imită un catalog de bibliotecă. Aplicația te va ajuta să ții evidența cărților dintr-o bibliotecă. Am dezvoltat un Express Application, pe partea de back-end am folosit limbajele GraphQL Server și NodeJS, pe partea de front-end am creat un client folosind React, pentru trimiterea cererilor am folosit un GraphQL client(Apollo), datele sunt stocate într-un database online(Atlas MongoDB).

TEHNOLOGIILE FOLOSITE



PACKAGE-URI FOLOSITE

Layer server

`npm install express`

`npm install nodemon -g`

`npm install graphql express-graphql`

`npm install lodash`

`npm install mongoose`

`npm install cors`

`npm install jsonwebtoken express-jwt`

Layer client

npm install -g create-react-app

**npm install apollo-boost @apollo/react-hooks
graphql**

npm install @apollo/client

npm install react-apollo

npm install lodash.flowright

SCREENSHOT-URI FACUTE ÎN TIMP CE DEZVOLTAM APLICAȚIA

Mai jos puteti găsi niste screenshot-uri în care am surprins unele dintre momentele cheie în dezvoltarea aplicației.

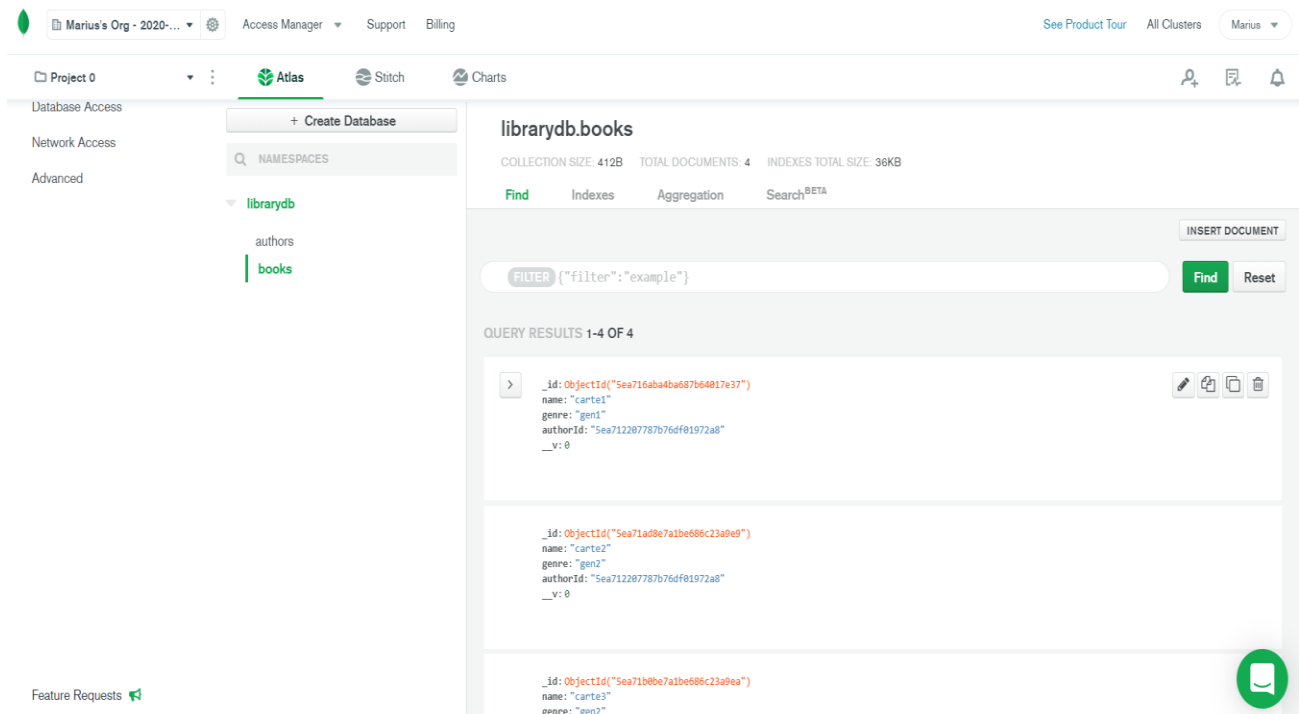
Configurare Atlas MongoDB

```
mongoose.connect('mongodb+srv://laurentiu:steaua123@librarycatalog-ezbff.mongodb.net/librarydb?retryWrites=true&w=majority', {
  useNewUrlParser:true,
  useUnifiedTopology:true
});

mongoose.connection.once('open',()=>{
  console.log('connected to db')
});
```

```
D:\ProiectWebSemantic\server>nodemon app
[nodemon] 2.0.3
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node app.js`
listening for requests on port 4000
connected to db
```

Primele date adăugate în DB folosind ca și client GrapiQL



The screenshot shows the MongoDB Atlas web interface. The top navigation bar includes 'Marius's Org - 2020-...', 'Access Manager', 'Support', and 'Billing'. The main header shows 'Project 0' and navigation icons for 'Atlas', 'Stitch', and 'Charts'. The left sidebar has 'Database Access', 'Network Access', and 'Advanced' sections. Under 'Database Access', there's a '+ Create Database' button and a search bar for 'NAMESPACES'. The 'librarydb' namespace is expanded, showing 'authors' and 'books' collections. The 'books' collection is selected, and the main panel displays 'librarydb.books' with 'COLLECTION SIZE: 412B', 'TOTAL DOCUMENTS: 4', and 'INDEXES TOTAL SIZE: 36KB'. Below this, there are tabs for 'Find', 'Indexes', 'Aggregation', and 'Search BETA'. A filter bar shows a filter query: `{ "filter": "example" }`. The 'QUERY RESULTS 1-4 OF 4' section displays three documents:

```
{ "_id": ObjectId("5ea716aba4ba687b64017e37"), "name": "carte1", "genre": "gen1", "authorId": "5ea712207787b76df01972a8", "__v": 0 }
```

```
{ "_id": ObjectId("5ea71ad8e7a1be686c23a9e9"), "name": "carte2", "genre": "gen2", "authorId": "5ea712207787b76df01972a8", "__v": 0 }
```

```
{ "_id": ObjectId("5ea71b0be7a1be686c23a9ea"), "name": "carte3", "genre": "gen2" }
```

Primul read la date în GrapiQL

GraphQL

Prettify

Merge

Copy

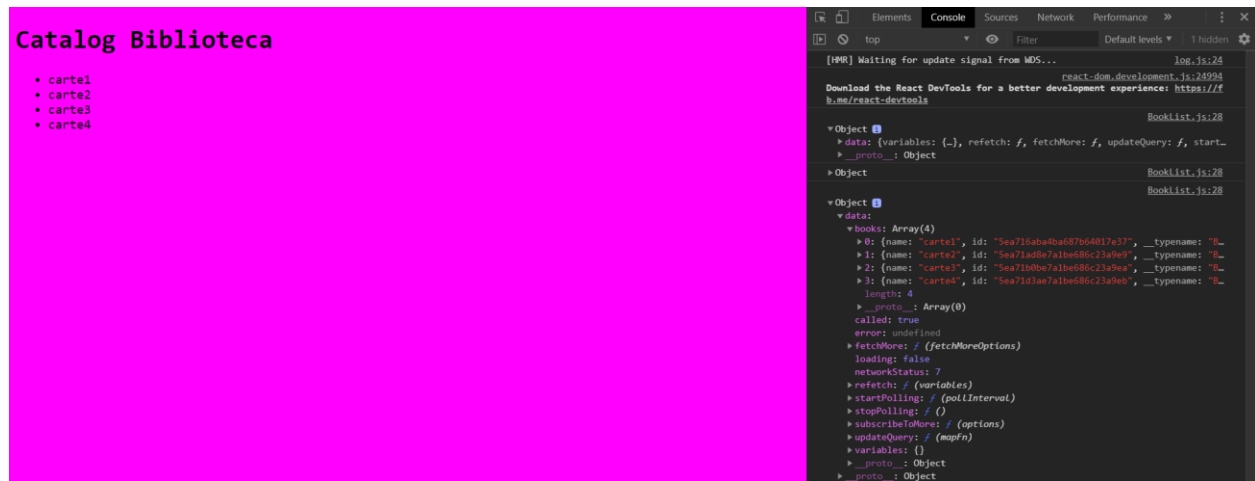
History

```
1 {
2   books{
3     name
4     genre
5     author{
6       name
7       age
8     }
9   }
10 }
```

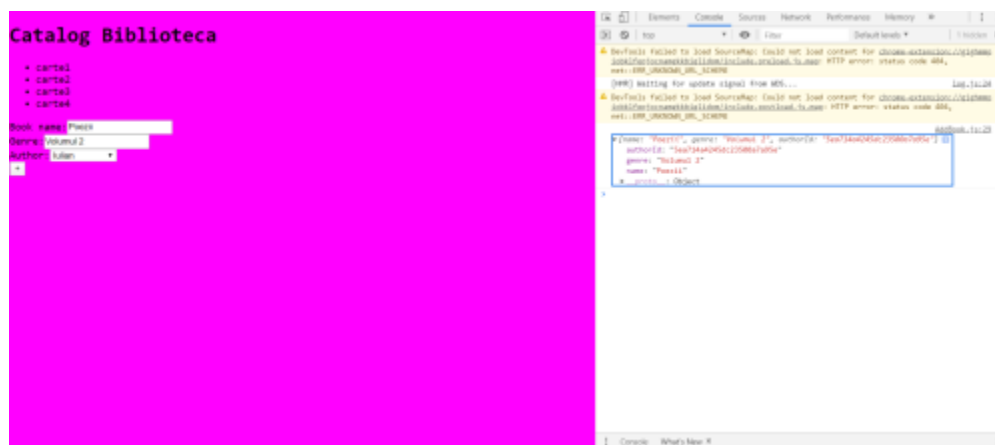
age
Int Self descriptive.

```
{
  "data": {
    "books": [
      {
        "name": "carte1",
        "genre": "gen1",
        "author": {
          "name": "Laurentiu",
          "age": 30
        }
      },
      {
        "name": "carte2",
        "genre": "gen2",
        "author": {
          "name": "Laurentiu",
          "age": 30
        }
      },
      {
        "name": "carte3",
        "genre": "gen2",
        "author": {
          "name": "Laurentiu",
          "age": 30
        }
      },
      {
        "name": "carte4",
        "genre": "gen2",
        "author": {
          "name": "Iulian",
          "age": 30
        }
      }
    ]
  }
}
```

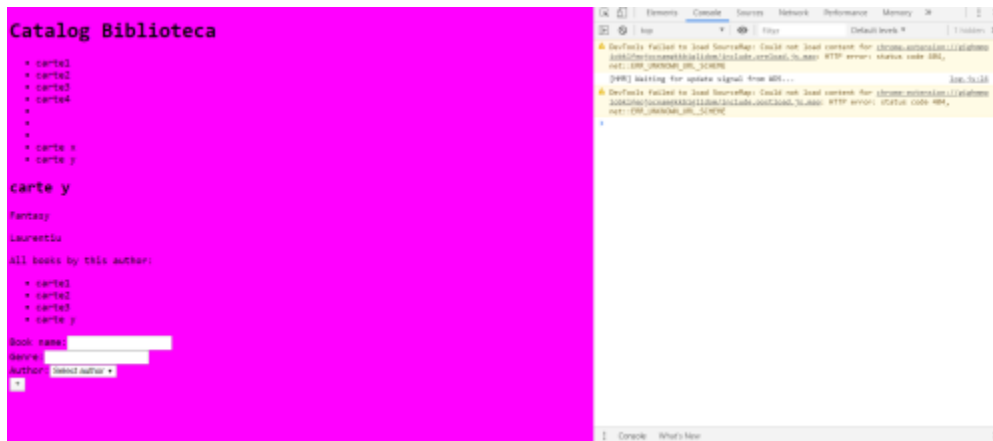

Primul read la date în React(Zona A)



Primul create din React(Zona C)



Al doilea read la date în React(Zona B)



Produsul final



BIBLIOGRAFIE

<https://graphql.org/learn/>

<https://docs.atlas.mongodb.com/getting-started/>

<https://github.com/facebook/create-react-app>

<https://www.apollographql.com/docs/react/get-started/>

https://www.w3schools.com/cssref/css_colors.asp

https://www.w3schools.com/cssref/css_websafe_fonts.asp