

CS 499 Capstone Narrative – Databases

Student Name: Emireth Castro

Course: CS 499 Computer Science Capstone

Enhancement: Migrating Rescue Animal Data from Static Files to MongoDB with Mongoose

A. Describe the Artifact

The original artifact used text files (`animals.txt`, `dogs.txt`, etc.) and in-memory data structures in Java to store and retrieve animal information. This made it difficult to scale or preserve data between program executions. Every time the Java console app restarted, all records were lost.

This enhancement transitioned the system into a NoSQL data architecture using MongoDB and Mongoose within a Node.js backend. It allowed for persistent data storage, schema validation, and real-time CRUD operations.

B. Justify the Inclusion

I selected this enhancement to demonstrate mastery in full-stack development, specifically backend database integration and validation. It represents a major advancement in how application data is stored, secured, and retrieved. Rather than using arrays or manual file reads/writes, the application now connects to a MongoDB instance using the Mongoose ORM.

The updated system includes:

- Two Mongoose models: `Dog.js` and `Monkey.js`
- Required fields and schema validation (e.g., age format, species type, date validation)
- MongoDB connection through environment-based URI (MongoDB Atlas/local)
- RESTful endpoints for:
 - `GET /dogs`, `GET /monkeys`
 - `POST /dogs`, `POST /monkeys`
 - `PUT /dogs/:id`, `PUT /monkeys/:id`
 - `DELETE` (optional)
- Filtering support using MongoDB query operators (e.g., `reserved=false`)
- Unit tests that insert, retrieve, and assert object structure

This enhancement shows that I can design real-world, persistent data layers and apply modern database interaction patterns using asynchronous Node.js logic.

C. Reflect on the Enhancement Process

Setting up MongoDB, defining schemas, and integrating CRUD routes required careful planning and testing. I had to learn how to:

- Connect securely using `.env` configurations
- Validate incoming data before writing to the database
- Design schemas that prevented incorrect values from being stored

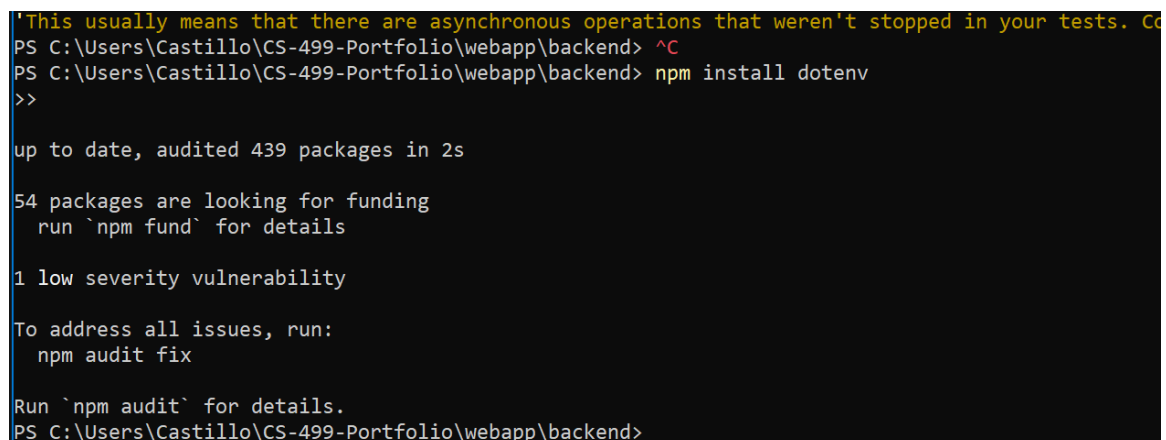
Using Mongoose helped abstract much of the low-level MongoDB complexity, but I still had to understand collections, indexes, and model lifecycle hooks.

I added unit tests to verify that animals inserted via POST requests matched the expected object shape and were successfully retrieved using GET. This gave me confidence in the persistence and structure of my application.

This enhancement aligns with course outcomes for managing data, designing secure and scalable backend systems, and applying validation and testing to database interactions.

Screenshots

1. Connection logic in `server.js` or `dbConnect.js`



```
'This usually means that there are asynchronous operations that weren't stopped in your tests. Co
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> ^C
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> npm install dotenv
>>

up to date, audited 439 packages in 2s

54 packages are looking for funding
  run `npm fund` for details

1 low severity vulnerability

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend>
```

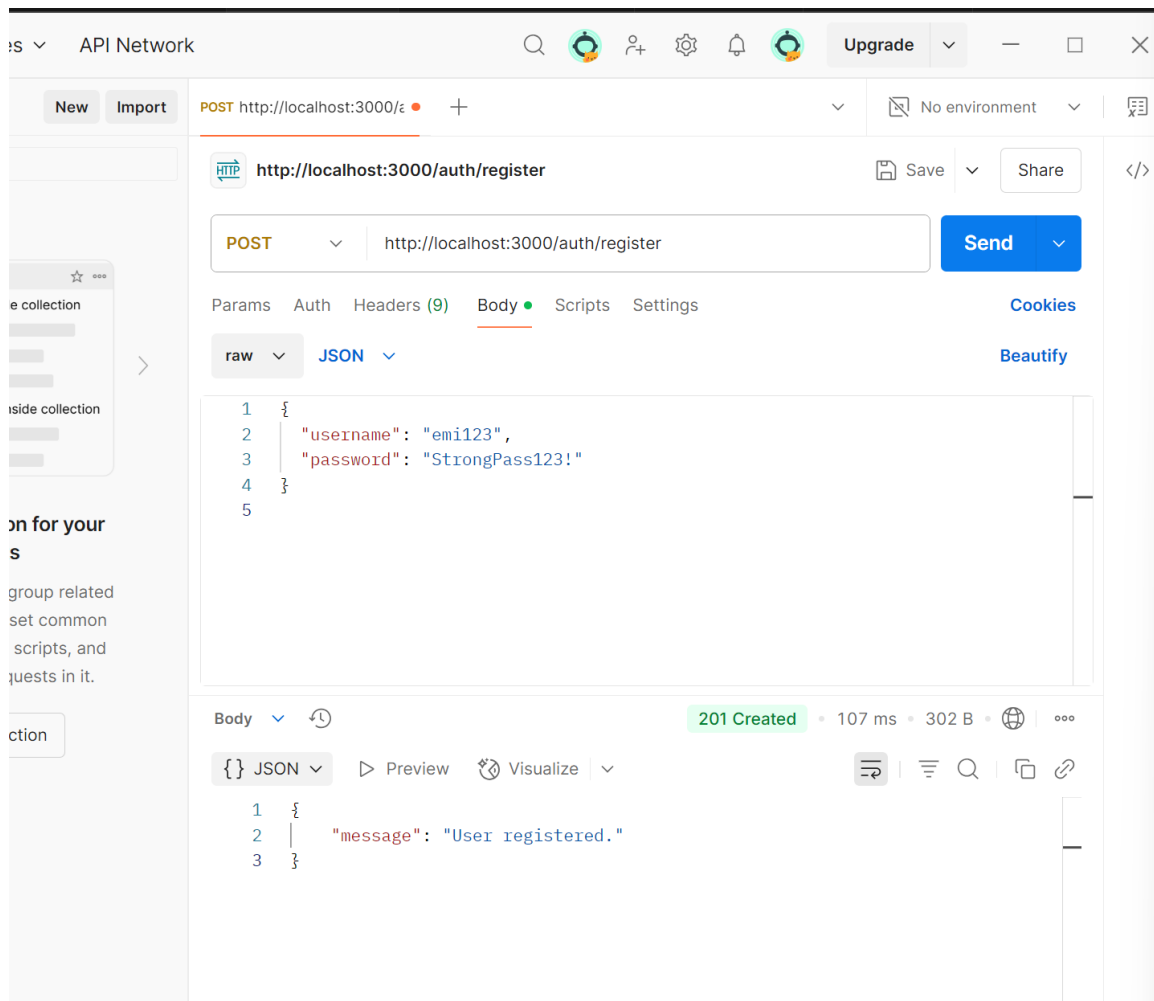
```
Ran all test suites.  
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> npm install bcrypt zxcvbn  
  
up to date, audited 444 packages in 2s  
  
54 packages are looking for funding  
  run `npm fund` for details  
  
1 low severity vulnerability  
  
To address all issues, run:  
  npm audit fix  
  
Run `npm audit` for details.  
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> npm test -- --detectOpenHandles
```

```
Ran all test suites.  
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> npm install module-alias  
  
added 1 package, and audited 445 packages in 1s  
  
54 packages are looking for funding  
  run `npm fund` for details  
  
1 low severity vulnerability  
  
To address all issues, run:  
  npm audit fix  
  
Run `npm audit` for details.  
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> |
```

```
Run `npm audit` for details.  
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> npm install  
  
up to date, audited 445 packages in 1s  
  
54 packages are looking for funding  
  run `npm fund` for details  
  
1 low severity vulnerability  
  
To address all issues, run:  
  npm audit fix  
  
Run `npm audit` for details.  
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> |
```

```
Run `npm audit` for details.  
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> npm test  
  
> grazioso-backend@1.0.0 test  
> cross-env NODE_ENV=test jest
```

2. Successful POST and GET requests in Postman



API Network

NewImport

POST http://localhost:3000/ε +

No environment

HTTP

http://localhost:3000/auth/register

Save

Share

</>

POST

http://localhost:3000/auth/register

Send

ParamsAuthHeaders (9)BodyScriptsSettings

Cookies

Beautify

rawJSON

```
1 {
2   "username": "emi123",
3   "password": "StrongPass123!"
4 }
5
```

Body

201 Created • 107 ms • 302 B •

JSON

PreviewVisualize

```
1 {
2   "message": "User registered."
3 }
```

API Network

NewImport

POST http://localhost:3000/ε +

No environment

HTTP

http://localhost:3000/auth/register

Save

Share

</>

POST

http://localhost:3000/auth/register

Send

ParamsAuthHeaders (9)BodyScriptsSettings

Cookies

Beautify

rawJSON

```
1 {
2   "username": "emi123",
3   "password": "StrongPass123!"
4 }
5
```

Body

201 Created • 107 ms • 302 B •

JSON

PreviewVisualize

```
1 {
2   "message": "User registered."
3 }
```

GET http://localhost:3000/m

No environment

HTTP

http://localhost:3000/monkeys/search?species=capuchin&reserved=false

Save

Share

GET

http://localhost:3000/monkeys/search?species=capuchin&reserved=false

Send

Params

Authorization

Headers (9)

Body

Scripts

Settings

Cookies

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

JSON

Beautify

```
1 {
2   "username": "emi123",
3   "password": "StrongPass123!"
4 }
5
```

Body

Cookies

Headers (8)

Test Results

200 OK

10 ms

533 B

🌐

⋮

{}

JSON

Preview

Visualize

```
1 {
2   "count": 1,
3   "results": [
4     {
5       "name": "Kiki",
6       "species": "Capuchin",
7       "gender": "female",
8       "age": "5",
9       "weight": "15",
10      "tailLength": "30",
11      "height": "45",
12      "bodyLength": "50",
13      "acqDate": "03-10-2021",
14      "acqCountry": "Brazil",
15      "trainingStatus": "in service",
16      "reserved": "false",
```

GET http://localhost:3000/dogs

No environment

HTTP

http://localhost:3000/dogs/search?status=complete&reserved=true

Save

Share

</>

GET

http://localhost:3000/dogs/search?status=complete&reserved=true

Send

Params

Authorization

Headers (9)

Body

Scripts

Settings

Cookies

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

JSON

Beautify

```
1 {
2   "username": "emi123",
3   "password": "StrongPass123!"
4 }
5
```

Body

Cookies

Headers (8)

Test Results

200 OK

10 ms

291 B

🌐

⋮

{ } JSON

Preview

Visualize

⌵

```
1 {
2   "count": 0,
3   "results": []
4 }
```


GET http://localhost:3000/dogs

No environment

HTTP

http://localhost:3000/dogs/search?status=in%20service&reserved=false

Save

Share

GET

http://localhost:3000/dogs/search?status=in%20service&reserved=false

Send

Params

Authorization

Headers (9)

Body

Scripts

Settings

Cookies

☐ none

☐ form-data

☐ x-www-form-urlencoded

☒ raw

☐ binary

☐ GraphQL

JSON

Beautify

```
1 {
2   "username": "emi123",
3   "password": "StrongPass123!"
4 }
5
```

Body

Cookies

Headers (8)

Test Results

200 OK • 44 ms • 474 B •

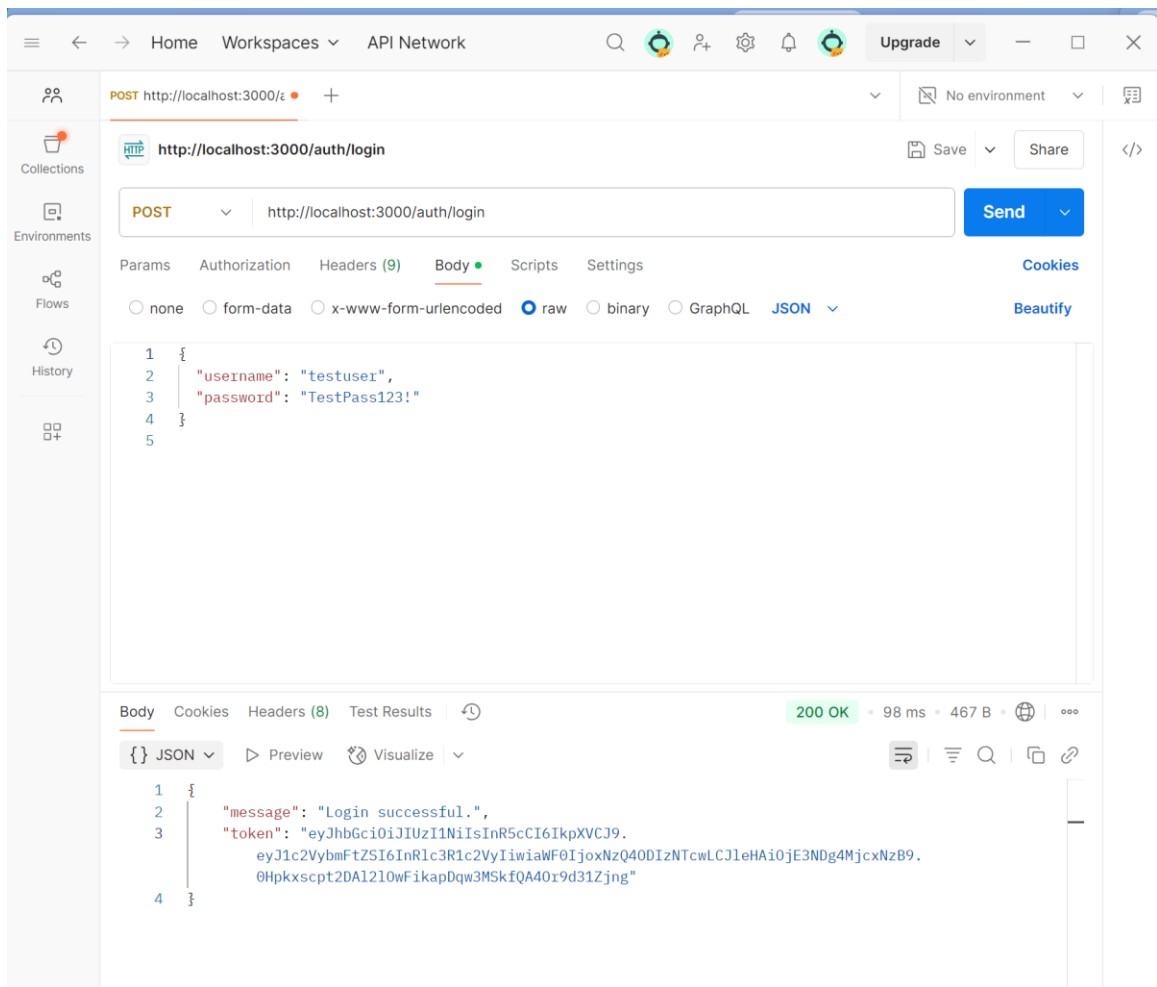
{}

JSON

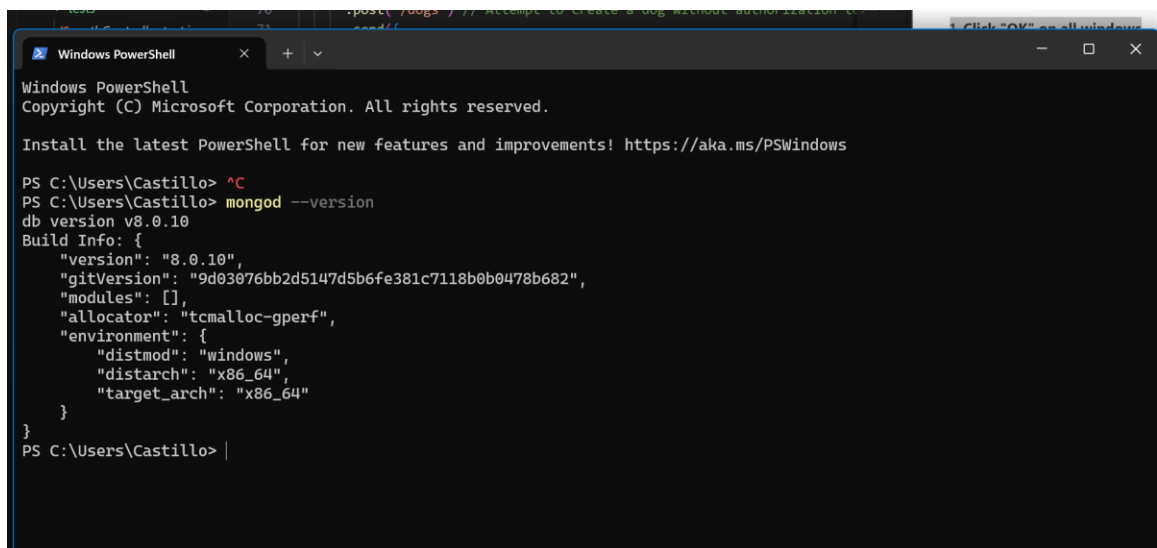
Preview

Visualize

```
1 {
2   "count": 1,
3   "results": [
4     {
5       "name": "Rex",
6       "breed": "Labrador",
7       "gender": "male",
8       "age": "4",
9       "weight": "30",
10      "acqDate": "01-01-2022",
11      "acqCountry": "USA",
12      "trainingStatus": "in service",
13      "reserved": "false",
14      "inService": "USA"
15    }
16  ]
17 }
```



3. MongoDB Compass view showing stored records



4. Jest test output showing successful record insertion and retrieval

```
Windows PowerShell
Time: 7.212 s
Ran all test suites.
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> npm test

> grazioso-backend@1.0.0 test
> cross-env NODE_ENV=test jest --config=jest.config.js

(node:16296) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
(node:16296) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
console.log
  ✓ MongoDB connected to mongodb://127.0.0.1:27017/grazioso_test
    at Object.connectTestDb (utils/testDbConnect.js:44:13)

POST /monkeys 400 9.066 ms - 49
POST /monkeys 400 1.943 ms - 106
POST /monkeys 201 51.829 ms - 432
POST /monkeys 409 8.194 ms - 31
POST /monkeys 201 18.468 ms - 429
POST /monkeys 401 0.640 ms - 47
console.log
  ✖ MongoDB connection closed
    at disconnectTestDb (utils/testDbConnect.js:59:13)

PASS tests/monkeyPost.test.js
  POST /monkeys
    ✓ should return 400 if required fields are missing (158 ms)
    ✓ should reject invalid species or taillength (26 ms)
    ✓ should reject duplicate monkey ID (103 ms)
    ✓ should create a new monkey successfully (38 ms)
    ✓ should reject request without a token (24 ms)

(node:16296) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
(node:16296) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
console.log
  ✓ MongoDB connected to mongodb://127.0.0.1:27017/grazioso_test
    at connectTestDb (utils/testDbConnect.js:44:13)

GET /dogs/search?id=D001 200 24.745 ms - 379
```

```
Windows PowerShell

    at connectTestDb (utils/testDbConnect.js:44:13)

GET /dogs/search?id=D001 200 24.745 ms - 379
GET /dogs/search?id=Z999 404 7.367 ms - 41
GET /dogs/search?name=lu 200 5.484 ms - 367
GET /dogs/search?trainingStatus=intensive 200 9.796 ms - 379
GET /dogs/search?reserved=true 200 11.111 ms - 367
GET /dogs/search?id=D001 401 1.547 ms - 47
  console.log
    MongoDB connection closed

    at disconnectTestDb (utils/testDbConnect.js:59:13)

PASS tests/dogSearch.test.js
  GET /dogs/search
    ✓ should find a dog by ID using binary search (135 ms)
    ✓ should return 404 for non-existent ID (51 ms)
    ✓ should filter dogs by partial name (fallback) (49 ms)
    ✓ should filter dogs by trainingStatus (fallback) (41 ms)
    ✓ should filter dogs by reserved=true (fallback) (50 ms)
    ✓ should reject unauthorized requests (missing token) (40 ms)

(node:16296) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
(node:16296) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since Node.js Driver version 4.0.0 and will be removed in the next major version
  console.log
    ✓ MongoDB connected to mongodb://127.0.0.1:27017/grazioso_test

    at connectTestDb (utils/testDbConnect.js:44:13)

POST /dogs 400 2.712 ms - 40
POST /dogs 400 1.744 ms - 55
POST /dogs 400 1.509 ms - 46
POST /dogs 201 23.865 ms - 384
POST /dogs 400 12.920 ms - 28
POST /dogs 201 19.083 ms - 382
POST /dogs 401 0.598 ms - 47
  console.log
    MongoDB connection closed

    at disconnectTestDb (utils/testDbConnect.js:59:13)

PASS tests/dogPost.test.js
  POST /dogs
```

```
Windows PowerShell
at disconnectTestDb (utils/testDbConnect.js:59:13)

PASS tests/dogPost.test.js
POST /dogs
  ✓ should return 400 if required fields are missing (100 ms)
  ✓ should reject invalid gender or trainingStatus (22 ms)
  ✓ should reject negative age or weight (26 ms)
  ✓ should reject duplicate dog ID (79 ms)
  ✓ should create a new dog successfully (42 ms)
  ✓ should reject dog creation without token (22 ms)

(node:16296) [MONGODB DRIVER] Warning: useNewUrlParser is a deprecated option: useNewUrlParser has no effect since Node.
js Driver version 4.0.0 and will be removed in the next major version
(node:16296) [MONGODB DRIVER] Warning: useUnifiedTopology is a deprecated option: useUnifiedTopology has no effect since
Node.js Driver version 4.0.0 and will be removed in the next major version
  console.log
    ✓ MongoDB connected to mongodb://127.0.0.1:27017/grazioso_test

    at connectTestDb (utils/testDbConnect.js:44:13)

GET /monkeys/search?id=M001 200 10.728 ms - 417
GET /monkeys/search?id=Z999 404 7.690 ms - 28
GET /monkeys/search?name=bo 200 5.602 ms - 416
GET /monkeys/search?reserved=true 200 8.369 ms - 416
GET /monkeys/search?id=M001 401 0.753 ms - 47
  console.log
    ✖ MongoDB connection closed

    at disconnectTestDb (utils/testDbConnect.js:59:13)

PASS tests/monkeySearch.test.js
GET /monkeys/search
  ✓ should return a monkey by ID using binary search (91 ms)
  ✓ should return 404 for a non-existent monkey ID (47 ms)
  ✓ should filter monkeys by partial name (43 ms)
  ✓ should filter monkeys by reserved=true (49 ms)
  ✓ should reject unauthorized requests (41 ms)

PASS tests/authController.test.js
  ✓ authController loads (3 ms)

  console.warn
    Invalid input to binarySearch

       9 | function binarySearch(arr, targetId) {
      10 |   if (!Array.isArray(arr) || typeof targetId !== 'string') {
      11 |     console.warn('Invalid input to binarySearch')
```

```
Windows PowerShell

at disconnectTestDb (utils/testDbConnect.js:59:13)

PASS tests/monkeySearch.test.js
  GET /monkeys/search
    ✓ should return a monkey by ID using binary search (91 ms)
    ✓ should return 404 for a non-existent monkey ID (47 ms)
    ✓ should filter monkeys by partial name (43 ms)
    ✓ should filter monkeys by reserved=true (49 ms)
    ✓ should reject unauthorized requests (41 ms)

PASS tests/authController.test.js
  ✓ authController loads (3 ms)

  console.warn
    Invalid input to binarySearch

       9 | function binarySearch(arr, targetId) {
      10 |   if (!Array.isArray(arr) || typeof targetId !== 'string') {
    > 11 |     console.warn("Invalid input to binarySearch");
        |               ^
      12 |     return null;
      13 |   }
      14 | }

at binarySearch (utils/binarySearch.js:11:13)
at Object.<anonymous> (tests/binarySearch.test.js:55:20)

PASS tests/binarySearch.test.js
  binarySearch()
    ✓ returns correct object when ID is in the middle (1 ms)
    ✓ returns correct object when ID is the first element (1 ms)
    ✓ returns correct object when ID is the last element (3 ms)
    ✓ returns null when ID does not exist (2 ms)
    ✓ returns null when input array is empty (1 ms)
    ✓ works with numeric-looking string IDs (1 ms)
    ✓ is case-insensitive with mixed casing (1 ms)
    ✓ returns null for non-string targetId (7 ms)
    ✓ skips invalid entries safely (2 ms)

Test Suites: 6 passed, 6 total
Tests:       32 passed, 32 total
Snapshots:   0 total
Time:        6.728 s
Ran all test suites.
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> |
```

5. GitHub branch view: database-enhancement

```
e87a933..2da7fe1 algorithms-data-structure-enhancement -> algorithms-data-structure-enhancement
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> git push origin algorithms-data-structure-enhancement
Everything up-to-date
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> git checkout -b database-enhancement
>>
Switched to a new branch 'database-enhancement'
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> git push -u origin database-enhancement
>>
Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote:
remote: Create a pull request for 'database-enhancement' on GitHub by visiting:
remote:   https://github.com/delfin7emi/CS-499-Portfolio/pull/new/database-enhancement
remote:
To https://github.com/delfin7emi/CS-499-Portfolio.git
 * [new branch]      database-enhancement -> database-enhancement
branch 'database-enhancement' set up to track 'origin/database-enhancement'.
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> npm install mongoose
>>

added 20 packages, and audited 439 packages in 6s

54 packages are looking for funding
  run `npm fund` for details

1 low severity vulnerability

To address all issues, run:
  npm audit fix

Run `npm audit` for details.
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> _
```

```
Windows PowerShell
PS C:\Users\Castillo\CS-499-Portfolio\webapp\backend> cd "C:\Users\Castillo\CS-499-Portfolio"
PS C:\Users\Castillo\CS-499-Portfolio> git checkout database-enhancement
D    webapp/backend/authMiddleware.js
M    webapp/backend/package-lock.json
M    webapp/backend/package.json
M    webapp/backend/server.js
M    webapp/backend/tests/authController.test.js
M    webapp/backend/tests/dogSearch.test.js
M    webapp/backend/tests/monkeyPost.test.js
M    webapp/backend/tests/monkeySearch.test.js
M    webapp/backend/users.json
Already on 'database-enhancement'
Your branch is up to date with 'origin/database-enhancement'.
PS C:\Users\Castillo\CS-499-Portfolio> git add .
warning: in the working copy of 'webapp/backend/users.json', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Castillo\CS-499-Portfolio> git commit -m "Final push for database enhancement - CS499"
[database-enhancement 03cc9ae] Final push for database enhancement - CS499
21 files changed, 1918 insertions(+), 391 deletions(-)
create mode 100644 webapp/.gitattributes
create mode 100644 webapp/backend/jest.config.js
create mode 100644 webapp/backend/tests/dogPost.test.js
create mode 100644 webapp/backend/tests/test.js
rename webapp/{backend => middleware}/authMiddleware.js (100%)
create mode 100644 webapp/middleware/validateDog.js
create mode 100644 webapp/middleware/validateDog.test.js
create mode 100644 webapp/middleware/validateMonkey.js
create mode 100644 webapp/middleware/validateMonkey.test.js
create mode 100644 webapp/middleware/validationmiddleware.js
create mode 100644 webapp/models/Animals.js
create mode 100644 webapp/models/Dogs.js
create mode 100644 webapp/models/Monkeys.js
PS C:\Users\Castillo\CS-499-Portfolio> git push origin database-enhancement
Enumerating objects: 39, done.
Counting objects: 100% (39/39), done.
Delta compression using up to 8 threads
Compressing objects: 100% (26/26), done.
```