Below is a clear, step-by-step guide to add an endpoint for creating products. This guide will add both a form to input new product details and a POST endpoint to save the product to your MongoDB Atlas database. Follow each step exactly.

Step 1: Update the Products Route

1. **Open routes/products.js:**

In the file where you already handle product listing, add two new routes—one to display the form and one to process the form submission.

2. **Add the GET Route for the New Product Form:**

Place the following code in routes/products.js to render a form for creating a new product:

```
// GET /products/new - Display the new product form
router.get('/new', (req, res) => {
  res.render('products/new', { title: 'Create New Product' });
});
```

3. **Add the POST Route to Process the Form:**

Next, add the following POST endpoint in the same file to handle the form submission and create a new product in the database:

```
// POST /products - Create a new product in the database
router.post('/', async (req, res) => {
  try {
    const { name, description, price, imageUrl } = req.body;
    const product = new Product({ name, description, price, imageUrl });
    await product.save();
    // After saving, redirect to the products listing page
    res.redirect('/products');
  } catch (err) {
    console.error(err);
    res.status(500).send('Server error while creating product.');
  }
});
```

Note: Ensure your file imports the Product model at the top:

```
import Product from '../models/Product.js';
```

Step 2: Create the New Product Form View

1. **Create the New EJS Template:**

In your views/products folder, create a new file called new.ejs.

2. **Add the Form Markup:**

Paste the following code into new.ejs:

```
<%- include('../partials/header.ejs') %>
<div class="container">
  <h2><%= title %></h2>
  <form action="/products" method="POST">
    <div>
      <label for="name">Product Name:</label>
      <input type="text" id="name" name="name" required>
    </div>
    <div>
      <label for="description">Description:</label>
      <textarea id="description" name="description"></textarea>
    </div>
    <div>
      <label for="price">Price:</label>
      <input type="number" id="price" name="price" step="0.01" required>
    </div>
    <div>
      <label for="imageUrl">Image URL:</label>
      <input type="url" id="imageUrl" name="imageUrl">
    </div>
    <button type="submit">Create Product</button>
  </form>
</div>
<%- include('../partials/footer.ejs') %>
```

Tip: Make sure that the path in the include statements is correct relative to new.ejs.

Step 3: Verify Your Application Setup

1. **Confirm the Route Mounting:**

In your main application file (app.js), verify that the products router is mounted. It should have a line similar to:

```
app.use('/products', productsRouter);
```

This ensures that both the GET and POST routes are active.

2. **Start the Server:**

In your terminal, run:

```
node app.js
```

or use your preferred method (e.g., nodemon).

3. **Test the New Endpoint:**

• Open your browser and navigate to http://localhost:3000/products/new. You should see the form to create a new product.

• Fill in the details and submit the form. If successful, you should be redirected to your products listing page (make sure you have existing code to display products).

This guide adds an endpoint to create new products by displaying a form and processing the form submission to store the product data. Follow each step carefully, and if you run into any issues, feel free to ask for help. Happy coding!