

Transforming a MERN Stack App to Use DynamoDB

In this exercise, you will transform a MERN (MongoDB, Express, React, Node.js) stack application to use DynamoDB as its database. DynamoDB is a fully managed NoSQL database service provided by AWS that offers fast and predictable performance, seamless scalability, and built-in security features.

Prerequisites

Before starting this exercise, you should have a basic understanding of the MERN stack, JavaScript, and TypeScript. You should also have an AWS account and some familiarity with DynamoDB.

Procedure

To transform a MERN stack application to use DynamoDB, follow these general steps:

1. Analyze the MongoDB schema. Review the schema of the MongoDB database (available in `models/task.model.js` in the TasksApp backend directory) and identify the key entities that are represented in the database. This will help you design the schema for the DynamoDB table(s) that will replace the MongoDB collection(s).
2. Create the DynamoDB table. Use the AWS Management Console (or the DynamoDB JS SDK for extra challenge) to create the DynamoDB table that will hold the data for the application. Be sure to define the primary key(s) for the table based on the entities and relationships you identified in step 1.
3. Update the backend code. Modify the backend code of the application to use the AWS SDK to interact with the DynamoDB table instead of the MongoDB collection. The bulk of your work will be in the backend code, in `controllers/task.controller.js`. Replace the MongoDB-specific queries and operations with their DynamoDB equivalents, such as
 - ScanCommand: to replace the `Task.find` MongoDB operation (or use the QueryCommand if you want to be more efficient)
 - DeleteCommand: to replace the `Task.findByIdAndRemove` MongoDB

operation

- UpdateCommand: to replace the `Task.findByIdAndUpdate` MondoDB operation
- PutCommand: to replace the `task.save` MondoDB operation

4. You don't need to work on the frontend code if you maintain the same API endpoints from your backend.
5. Test the updated application. Test the updated application to ensure that it functions correctly and provides the same features and functionality as the original MERN stack application.

The documentation for the AWS SDK for JavaScript V3 can be found [here](#)