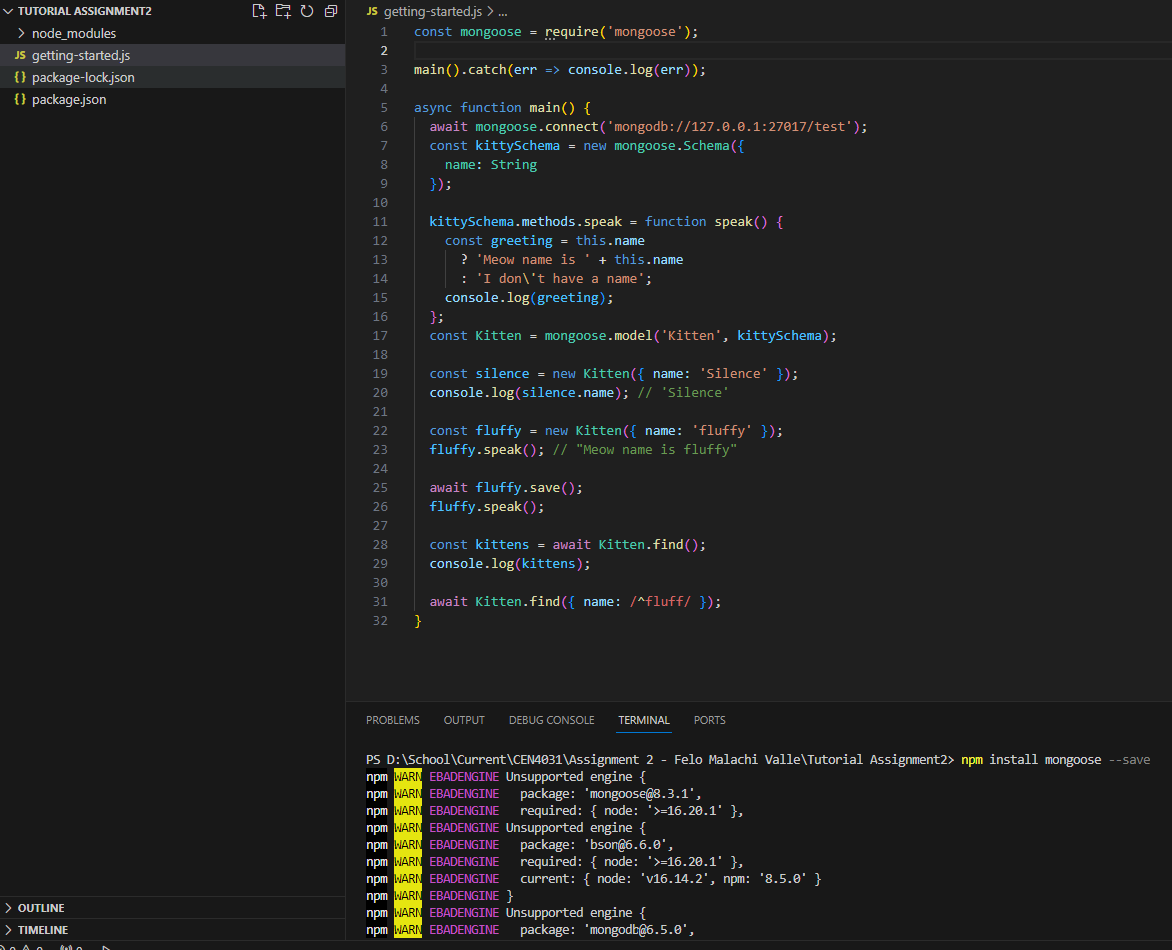
Project Report

\*\*MongoDB: \*\*

MongoDB is a framework used to create and manage databases on internet applications while remaining highly available and scalable. It utilizes schemas to organize data structures and dictate how the data is stored. It is like SQL; however, records are stored in a BSON document, which can be retrieved as JSON. The biggest advantage of MongoDB is its scalability, and its flexibility with how the data can be stored.



\*\*Support Features: \*\*  
- Does your framework support batching? If so, how?

Yes, it does support batching. The best implementation of batching is to store similar data in embedded documents, and always fetch related data together.

- Does your framework have support for connection pooling? If so, how?

Yes, MongoDB supports connection pooling, and this is present in all MongoDB drivers. These connections are managed automatically by the drivers, each connection is stored in a MongoClient object, and these connections only last during the application’s lifetime.

- Does your framework have support for exception management? If so, how?

No, there is not targeted support for MongoDB, however, MongoDB is implemented through JavaScript, so JavaScript exception handling can be applied to MongoDB through all JavaScript techniques (try catch blocks, etc.).

- Does your framework have support for queries? If so, how?

Yes, MongoDB can query the database using MQL or aggregation. Aggregation is preferred because more precise results can be filtered down, versus using MQL.

- Does your framework support validation? If so, how?

Yes, MongoDB applies validation checks to all documents by default. It will also validate all schemas.

- Does your framework support transactions? If so, how?

Yes, MongoDB supports distributed transactions. Distributed transactions can be used to record multiple operations, collection, and even the databases.

\*\*Data Access Patterns: \*\*  
- What pattern does your framework promote for data access? (Active Record, Data Mapper, Data Transfer Object, Domain Model, Query Object, Repository, Row Data Gateway, Table Data Gateway, Table Module)

MongoDB uses Query Object for data access. When data is needed, MongoDB will utilize an aggregate function on an object, created from the BSON file.

- What Domain Model Pattern does your framework provide support for? (Active Record, Domain Model, Transaction Script, Table Module)

MongoDB supports Domain Model, because each record is predefined, like an object.

\*\*Framework Evaluation: \*\*  
Personally, MongoDB was much harder for me to implement, but I have a bias towards any backend/database development. I much prefer front-end developing frameworks, and the visual aspect of the code. In general, MongoDB is widely used due to its flexibility, scalability, and being a JavaScript based framework, it eliminates the need for SQL.