Period-2 Node, Express with TypeScript, JavaScript Backend Testing, MongoDB (and Geo-location)

Note: This description is too big for a single exam-question. It will be divided up into separate questions for the exam

Explain Pros & Cons in using Node.js + Express to implement your Backend compared to a strategy using, for example, Java/JAX-RS/Tomcat

Explain the difference between *Debug outputs* and *ApplicationLogging*. What's wrong with console.log(..) statements in our backend code?

Demonstrate a system using application logging and environment controlled debug statements.

Explain, using relevant examples, concepts related to testing a REST-API using Node/JavaScript/Typescript + relevant packages

Explain a setup for Express/Node/Test/Mongo-DB/GraphQL development with Typescript, and how it handles "secret values", debug and testing.

Explain, preferably using an example, how you have deployed your node/Express applications, and which of the Express Production best practices you have followed.

Explain possible steps to deploy many node/Express servers on the same droplet, how to deploy the code and how to ensure servers will continue to operate, even after a droplet restart.

Explain, your chosen strategy to deploy a Node/Express application including how to solve the following deployment problems:

- Ensure that you Node-process restarts after a (potential) exception that closed the application
- Ensure that you Node-process restarts after a server (Ubuntu) restart
- Ensure that you can run "many" node-applications on a single droplet on the same port (80)

Explain, using relevant examples, the Express concept; middleware.

Explain, using relevant examples, your strategy for implementing a REST-API with Node/Express + TypeScript and demonstrate how you have tested the API.

Explain, using relevant examples, how to test JavaScript/Typescript Backend Code, relevant packages (Mocha, Chai etc.) and how to test asynchronous code.

NoSQL and MongoDB

Explain, generally, what is meant by a NoSQL database.

Explain Pros & Cons in using a NoSQL database like MongoDB as your data store, compared to a traditional Relational SQL Database like MySQL.

Explain about indexes in MongoDB, how to create them, and *demonstrate* how you have used them.

Explain, *using your own code* examples, how you have used some of MongoDB's "special" indexes like *TTL* and *2dsphere and perhaps also the Unique Index.*

Demonstrate, using a REST-API *designed by you*, how to perform all CRUD operations on a MongoDB

Explain, using a *relevant example*, a full JavaScript backend including relevant test cases to test the REST-API (not on the production database)

Demonstrate, using your own code-samples, decisions you have made regarding → normalization vs denormalization

Geo-location and Geojson (Period-4)

Explain and demonstrate basic Geo-JSON, involving as a minimum, Points and Polygons

Explain and demonstrate ways to create Geo-JSON test data

Explain the typical order of longitude and latitude used by Server-Side APIs and Client-Side APIs

Explain and demonstrate a REST API that implements geo-features, using a relevant geo-library and plain JavaScript

Explain and demonstrate a REST API that implements geo-features, using Mongodb's geospatial queries and indexes.

Explain and demonstrate how you have tested the gameFacade and gameAPI for the game-related parts of the period exercises

This will come in period-5

Explain and demonstrate a React Native Client that uses geo-components (Location, MapView, etc.)

Explain and demonstrate both server and client-side, of the geo-related parts of your implementation of the ongoing semester case.