Period-4 Geo-location on the backend (GeoJson, MongoDB and geoqueries)

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

MongoDB indexes and Geo-features

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

"Indexes support the efficient execution of queries in MongoDB. Without indexes, MongoDB must perform a collection scan*, i.e. scan every document in a collection, to select those documents that match the query statement. If an appropriate index exists for a query, MongoDB can use the index to limit the number of documents it must inspect." (mongo db website)*

Example: fullstack-startcode/test/positionFacadeTest.ts
Example: https://account.mongodb.com/account/login

(og setupTestPositions)

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.

A 2dsphere index supports queries that calculate geometries on an earth-like sphere.

The 2dsphere index supports data stored as GeoJSON objects *and* legacy coordinate pairs

Example: fullstack-startcode/src/facades/positionFacade.ts

Example: fullstack-startcode/test/positionFacadeTest.ts

TTL (Time To Live) indexes are special single-field indexes that MongoDB can use to automatically remove documents from a collection after a certain amount of time or at a specific clock time.

Example: fullstack-startcode/test/positionFacadeTest.ts

Example: fullstack-startcode/src/facades/positionFacade.ts

Geo-location and Geojson

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

Example: fullstack-startcode/src/utils/geoUtils.ts

Example: Period4/geo-start-noDB-main/src/gameData.js

Explain and demonstrate ways to create Geo-JSON test data

Since we have an in-memory database, we can create the index in the “before” method, since the database is terminated after test run.

Example: fullstack-startcode/test/positionFacadeTest.ts

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

lon, lat	lat, lon
formats • GeoJSON ref • KML ref • Shapefile ref • WKT ref • WKB ref • geobuf ref	formats • GeoRSS ref • Encoded Polylines (Google) ref
javascript apis • OpenLayers ref • d3 ref • ArcGIS API for JavaScript ref • Mapbox GL JS ref	javascript apis • Leaflet ref • Google Maps API ref
mobile apis • Tangram ES[1]	mobile apis • Google Maps iOS/Android • Apple MapKit
service specifications • WFS 1.0.0 [1] • WMS 1.1.1 [1]	service specifications • WFS 1.1.0 & 2.0.0 [1] • WMS 1.3.0 [1]
misc • OSRM[1] • Redis[1]	misc

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

Check out addPosition, and findnearby Friends in example.

Example: fullstack-startcode/src/facades/positionFacade.ts

Example: fullstack-startcode/src/graphql/schema.ts

Explain and demonstrate a GraphQL API that implements geo-features, using Mongodb's geospatial queries and indexes.
Basically entire code: fullstack-startcode/src

Explain and demonstrate how you have tested the geo-related features in you start code

Remember it is an in memory database.

Example: fullstack-startcode/test