**Written Questionnaire - Answers**

**1.Explaining what design pattern is and how we can use design patterns in projects.**

A Design pattern is a written description of a generic solution to a design problem that frequently arises in many projects in the field of software development. The pattern solution is adjusted by software developers for each individual project. A design problem, its suggested solution, and any other elements that might have an impact on the problem or the solution are all formalized in patterns. A successful pattern should have demonstrated itself in three prior projects or circumstances as producing a suitable answer.

A pattern in object-oriented programming may outline specific objects and object classes to be used, their characteristics and dependencies, as well as the general strategy for solving the issue. Programmers frequently have access to multiple patterns to solve a given issue. A pattern framework is a set of patterns.

**2. What is DTO and explain the use of it.**

Data Transfer Objects, or DTOs, are objects that transfer data between processes to reduce the method calls.

According to Fowler, the pattern's primary goal is to decrease the number of times a user must call the server by grouping up several parameters into a single call. As a result, network latency for such remote tasks is decreased.

The logic of serialization is also encapsulated, which is another advantage  Additionally, it separates the display layer from the domain models, enabling independent change for each.

DTOs are useful in systems that make remote calls since they help to reduce  on their frequency.DTOs can further reduce roundtrips between the client and server when the domain model is made up of numerous separate objects and the presentation model requires all of their data at once.

**3. How are you going to store secrets in an application without exposing it to the internet?**

I create .env file to store the database connection link. But I did not include it to this repository because of the security . I used .gitignore file to store secrets without exposing it to the internet

**4. What is JWT and how does it work?**

JWT, or JSON Web Token, is an open standard that allows a client and a server to communicate security-related data. Every JWT has a set of encoded JSON objects, including claims. To ensure that the claims cannot be changed after the token is issued, JWTs are signed using a cryptographic technique.

JWTs are unique among web tokens in that they include a list of claims. Information is sent between two parties through claims. The specific use case will determine what these assertions are. A claim might specify, for instance, who issued the token, how long it is valid, or what authorizations the client has received.

Three sections of a string called a JWT are separated by dots (.) and serialized with base64. The JWT appears something like this in compact serialization, the most used serialization format: xxxxx.yyyyy.zzzzz.

We can get two JSON strings, after decoding:

1. The payload and the header.
2. The signature.

The type of token, in this case a JWT, and the signing technique are both contained in the JOSE (JSON Object Signing and Encryption) header.

**5. What is the difference between SQL and NoSQL databases?**

* SQL databases are commonly referred to as RDBMS, or Relational Database Management Systems, and NoSQL databases are typically referred to as Non-relational or Distributed Databases.
* Traditional RDBMS analyze and retrieve data using SQL syntax and queries in order to gain additional insights. For OLAP systems, they are employed. A NoSQL database system is made up of different database technologies. These databases were created in response to the requirements put forward for the creation of current applications.
* SQL databases are table-based, but NoSQL databases might be document-based, key-value pair-based, or graph-based.
* While NoSQL databases are horizontally scalable, SQL databases are scalable vertically.
* While NoSQL databases use dynamic schema for unstructured data, SQL databases have a preset schema.
* When comparing the performance of SQL vs NoSQL, SQL requires specialized database hardware while NoSQL uses standard hardware.

**6. Suggest a good state management for frontend application and explain why you recommend it.**

I Suggest Redux for the state management for frontend application. However, it is also compatible with many other frameworks, including Angular, Vue, React, and standard JavaScript. It is most frequently used with ReactJS. It's crucial to remember that React and Redux are independent of one another despite frequently being used together.

Redux store for a centralized state management system. The data is easily accessible to all of the application's components directly. This centralizes all data and makes obtaining the state a component needs relatively simple for components. The Redux store is therefore strongly recommended when creating large, complex apps with different components.

It is frequently used to store long-term data needed as a user navigates an application, like data loaded from an API or data submitted through a form.

React updates every component in the component tree whenever one of the components is modified. These re-renders are useless in this situation because the data for a specific component hasn't changed. Redux store helps with efficiency by avoiding these pointless re-renders and guaranteeing that a given component only re-renders when its data has truly changed.