RESEARCH ASSIGNMENT

**Name: Hassan Majid**

**Roll.no: l164396**

Q1:

Mongoose is just a JavaScript library for MongoDB, which is a database, used to abstract and simplify some of the boilerplate in interaction with MongoDB, exposed by its native API, including object-modeling and introducing some level of schematic coherence.

In a nutshell - MongoDB is the software that manages durable storage and efficient retrieval & processing of your data, Mongoose is a set of high-level APIs for interaction with MongoDB, using JS as a language.

CREATE:-

db.student.insert({

rollNo: "4103",

name: "MSZ",

course: {

courseName: "WEB",

duration: "6 Months"

},

address: {

city: "Lahore",

state: "Punjab",

country: "Pakistan"

}

})

READ:-

db.students.find({"rollNo":"4396"})

UPDATE:-

db.student.update({

"rollNo": "3014"

},

$set:

{

"name":"Hassan Majid"

})

DELETE:-

db.collection\_name.remove({"state":"Punjab"})

Q2: PUT VS PATCH

Both PUT and POST can be used for creating.

You have to ask "what are you performing the action to?" to distinguish what you should be using. Let's assume you're designing an API for asking questions. If you want to use POST then you would do that to a list of questions. If you want to use PUT then you would do that to a particular question.

Great both can be used, so which one should I use in my RESTful design:

You do not need to support both PUT and POST.

Which is used is left up to you. But just remember to use the right one depending on what object you are referencing in the request.

Some considerations:

Do you name your URL objects you create explicitly, or let the server decide? If you name them then use PUT. If you let the server decide then use POST.PUT is idempotent, so if you PUT an object twice, it has no effect. This is a nice property, so I would use PUT when possible.

You can update or create a resource with PUT with the same object URL.With POST you can have 2 requests coming in at the same time making modifications to a URL, and they may update different parts of the object.

Q3:

POST and PUT can help consumers of your REST api understand what is happening in your API. For example you may require some kind of token on PUT (aka update) to help ensure that the entity being updated hasn't been changed since it was read. POST might fail differently when the entity already exists vs. PUT failing only if it has been changed or failing if it DOES NOT exist.

Q4:

ReactJS is an open-source, isomorphic JavaScript library but not a framework.It is developed by Facebook.As well as also seen on Instargram in 2012. It provides the possibility to create apps that are updated from time to time without the need to reload the page. The main purpose is to create high-performance solution that are updated from time to time without having to reload the page. This partial update technology is very convenient for single page apps developers. It uses one-way data binding with immutable data structures and give us both server-side and client-side rendering to give it a performance edge over competing technologies.

AngularJS analysis

Regarding AngularJS, it is very popular and robust framework with open source code for web applications that consist of one HTML-pages with CSS JavaScript named (SPAs). In fact, AngularJS is monolithic frameworks which include these three paradigms: Models, Views and Controllers, as we all known as MVC design patern. But Angular developers say that it actually isn’t MVC and more looks like MV\*. And perfectly, in our opinion, is coming to developments regarding small and medium projects. Although views in AngularJS has a very compact shape in terms of handling large amounts of data it loses in Angular vs React fight to the latter.

Q5:

Vue was created by Evan You after working for Google using AngularJS in a number of projects. He later summed up his thought process: "I figured, what if I could just extract the part that I really liked about Angular and build something really lightweight."[6] Vue was originally released in February 2014.Even though Vue is a relatively new technology, it is already being used in several large-scale projects mostly in China. The websites of Alibaba, Xiaomi, Baidu, and Tencent all use Vue.js as a central frontend technology.

Vue can be used for many different types of applications. Due to its compatibility with other JavaScript libraries and ability to add more complex logic to the existing apps, it can provide a perfect solution for nearly any type of project. But you should consider Vue.js as a primary option for the following requirements:

1 Dynamic high-performance applications: Similar to React, Vue.js is a good choice for dynamic applications, however thanks to virtual DOM it offers higher performance, which is beneficial for complex apps.

2. Single page apps: Vue.js allows changing the content quickly without reloading the page, which makes this framework perfect for SPAs.

Q6:

Angular was a ground-up rewrite of AngularJS.

1.Angular does not have a concept of "scope" or controllers, instead it uses a hierarchy of components as its primary architectural characteristic.

2.Angular has a different expression syntax, focusing on "[ ]" for property binding, and "( )" for event binding Modularity – much core functionality has moved to modules

3.Angular recommends the use of Microsoft's TypeScript language, which introduces the following features:

Class-based Object-Oriented Programming

Static Typing

Generics

4.TypeScript is a superset of ECMAScript 6 (ES6), and is backwards compatible with ECMAScript 5 (i.e.: JavaScript). Angular also includes ES6:

Lambdas

Iterators

For/Of loops

Python-style generators

Reflection

5.Dynamic loading

6.Asynchronous template compilation

Iterative callbacks provided by RxJS. RxJS limits state visibility and debugging, but these can be solved with reactive add-ons like ngReact or ngrx.

AngularJS

1.The original angularjs

2.More popular

3.Has more libraries and frameworks already established for it

4.Uses javascript

Q7:

Lint was the name originally given to a particular program that flagged some suspicious and non-portable constructs (likely to be bugs) in C language source code. The term is now applied generically to tools that flag suspicious usage in software written in any computer language.

Linting code is already an established part of any (popular) JavaScript project and, in my opinion, has a lot of benefits such as:

1.Readability

2.Pre-code review

3.Finding (syntax) errors before execution

ESLint is an open source, JavaScript linting utility originally created by Nicholas C. Zakas. Code linting is a type of static analysis that is frequently used to find problematic patterns or code that doesn’t adhere to certain style guidelines. There are code linters for most programming languages, and compilers can sometimes incorporate linting into the compilation process.

For further explanation, I will use the following simple script which generates an AST for the given JavaScript code.

var espree = require('espree');

var fs = require('fs');

var code = `let array = [1,2,'b'];

`;

var ast = espree.parse(code, {

ecmaVersion: 6

});

console.log("writing ast to ast.json")

fs.writeFile("ast.json", JSON.stringify(ast, null, 4), function(err) {

if(err) return err;

});

Q8:

Ajax is the feature which allows you to update the part of the page without update or refresh the page while AngularJS is one of the JavaScript framework (to be specific) client side MVC framework (most says MVW, where W is whatever that is MVC/MVVM )

Even before AngularJS ajax calls were used (today also) to update the particular part of a page and now you can use Angular for too for Ajax calls.One is feature or say way to attain a certain functionality while other is a big framework which include many other functionalities along with Ajax.

You can think of AJAX as the ability to get data from a server without the need to refresh a webpage.

However, Angular extends this idea with two-way data binding. So the HTML elements on your front end page are in constant communication with your back end server--and vice versa. Angular can achieve other things as well and is useful for separating your concerns, i.e. separating your data, the functions performed on that data, and how the user sees the data.