**Question 1**:

**Intent** : Learn the basics of how web applications are built and what technologies could be used for web applications.

Suppose you were given to build a web application for <https://www.boats.com/> from scratch as a startup . Imagine you would be solely responsible to build it’s frontend, backend, design database, setup media storage, deployment, ci/cd automation and everything else. Thus, do some research on your part and write down your response to the following questions.

1. **Which backend framework would you prefer to use and why?**

The main things we need to take into consideration is two perspectives,

1. Users experience
2. Developers experience

Selecting a backend framework depends on various factors like budget, needs of project. We will have to go through each feature of components that we are going to build so as to design the best framework for any given project

There are multiple backend frameworks, to name some:- Django, Flask, Express.js etc

For building this website I would use Express.js which is built on Node.js to build web applications due to the following reasons

* **Fast and scalable**- its built-on top of Google’s V8 JavaScript engine and it can handle large scale, real time applications in ease.
* **Scalability**- it can handle huge datasets and maintain data traffic
* **Cross-platform**- it can run on various operating systems such as windows, Linux, Mac OS etc.
* **Large echo system-** it allows for concurrent processing of multiple request without restricting the server.
* **Non-blocking Input/Output model-** it has a huge library of NPM which makes it easy to integrate with other technologies and services.

1. **Which frontend framework would you prefer to use and why?**

Choosing a frontend also depends on various factors like developer skills, developer’s creativity, application requirements etc.

For developing this particular application I would proceed with Next.js which is built on React, due to the following reasons

* **High performance**- Virtual Document Object Model(DOM) is used in React due to which enables it to perform very well even for applications with large number of components, this ensures efficient user experience
* **Reusability**- here developers will be able to reuse an already created component as and when needed.
* **Large community**- it has huge number of developers who are there to solve any query asked by a learner at any point of time
* **Easy-to-learn**: It is easily understood even by a fresher and is fun to learn
* **Compatibility-** it can be integrated with a wide range of technologies and platforms which includes backend framework, database etc.
* **Search engine optimization-** It makes sure that the website is among the top link when searched about it.

1. **Which database would you prefer to use and why?**

Choosing database depends on data complexity, ease of maintenance etc.

There are generally SQL and NO SQL databases

MySQL comes under SQL

SQL databases are used when we have relational data.

MongoDB comes under NO SQL

Here data is stored as a document.

So for this particular website we can use MongoDB as the database.

As we are designing a database for a startup, it will undergo many iteration.

Structure of the data/ schema will change multiple times and as data is not stored in a relational format it is better to go with MongoDB.

Features of MongoDB are:

* **Dynamic schema**- different types of data can be stored
* **Scalability**- it’s designed to scale horizontally, ie., can add more servers to handle data traffic and data storage needs.
* **Community support**- is very high, as many people are working on it, all queries will be answered soon!
* **Fields of application**- IoT, Mobile application, E-commerce, Content management system

1. **What would you use for version control of the codebase ?**

Git can be used for version control of the code base.

It is used to keep track of changes made to the source code and other files.

It is a powerful tool for version control that enables developers to work together on projects and keep track of changes over time. It is widely used in software development

1. **Which platform would you prefer for media storages ( eg. storage of product images) ?**

I would prefer S3 bucket (AWS)

As we are selecting a media storage component for a startup we need to keep the pricing in mind.

S3 bucket is an AWS platform and works on pay-as-you go model which makes it feasible and we can scale it up to our needs.

1. **Where would you deploy your application and which web server would you use and why?**

AWS EC2 instance provides us facilities to launch our servers

For deploying backend I would use EC2 instance

FOR IMNTERWIEW

1. **Git**

version control system that allows developers to manage and track changes to their codebase. It is widely used in software development to collaborate with other developers, track changes to code over time, and revert to previous versions of the codebase if necessary.

1. **GitHub**

web-based hosting service for Git repositories. It provides a user-friendly interface for managing Git repositories, as well as collaboration features such as pull requests, issue tracking, and wikis.

**Git and GitHub together allows developers to work collaboratively on code, share code with others, and keep track of changes to the codebase over time**

Need for version control-

* History tracking
* Collaboration
* Experimentation
* Reverting to previous versions
* Code review

1. Client server architecture

Client-frontend

Server-backend

Client request server, server searches in database sends response, client sends back acknowledgment telling it received the data

DNS domain name system- to change from host name to IP address

API- word Application refers to any software with a distinct function. Interface can be thought of as a contract of service between two applications. This contract defines how the two communicate with each other using requests and responses

A software intermediary that allows two applications to talk to each other

RESTful API is an interface that two computer systems use to exchange information securely over the internet.