



Advance Web Development

Mongo, Expressjs, Reactjs, Node

By Poga Kuofie

Topic	Introduction to MERN	
I can	I can define MERN	I can define the components of MERN
Vocabulary	React js, Node js, Components, Runtime, Cloud	
Assessment	Aptitude 1	



What is the MERN Stack?

MERN stands for MongoDB, Express, React, Node, after the four key technologies that make up the stack.

- MongoDB - document database
- Express(.js) - Node.js web framework
- React(.js) - a client-side JavaScript library
- Node(.js) - the premier JavaScript web server



Popularity & Uses

According to Stackoverflow 2020 Developer Survey, for the eighth year in a row, JavaScript has maintained its stronghold as the most commonly used programming language. Followed by HTML/CSS.

Reactjs is the second most popular framework after jQuery and its popularity is on the rise.

On Other Frameworks, Libraries, and Tools, for the second year in a row, Node.js takes the top spot, as it is used by half of the respondents.



Popularity & Uses

- Uber
- Paypal
- MTN Ghana
- Pulse Ghana
- Amalitech



Components Of MERN

How does the MERN stack work?

The MERN architecture allows you to easily construct a 3-tier architecture (frontend, backend, database) entirely using JavaScript and JSON.



Components Of MERN

React.js Front End

The top tier of the MERN stack is React.js, the declarative JavaScript framework for creating dynamic client-side applications in HTML. React lets you build up complex interfaces through simple Components, connect them to data on your backend server, and render them as HTML.



Components Of MERN

Express.js and Node.js Server Tier

The next level down is the Express.js server-side framework, running inside a Node.js server. Express.js bills itself as a “fast, unopinionated, minimalist web framework for Node.js,” and that is indeed exactly what it is. Express.js has powerful models for URL routing (matching an incoming URL with a server function), and handling HTTP requests and responses.



Components Of MERN

MongoDB Database Tier

If your application stores any data (user profiles, content, comments, uploads, events, etc.), then you're going to want a database that's just as easy to work with as React, Express, and Node.

That's where MongoDB comes in: JSON documents created in your React.js front end can be sent to the Express.js server, where they can be processed and (assuming they're valid) stored directly in MongoDB for later retrieval.



Why MERN stack?

MongoDB works extremely well with Node.js, and makes storing, manipulating, and representing JSON data at every tier of your application incredibly easy.

Express.js (running on Node.js) and React.js make the JavaScript/JSON application MERN full stack, well, full. Express.js is a server-side application framework that wraps HTTP requests and responses, and makes it easy to map URLs to server-side functions. React.js is a frontend JavaScript framework for building interactive user interfaces in HTML, and communicating with a remote server.



MERN Use Cases

Like any web stack, you can build whatever you want in MERN - though it's ideally suited for cases that are JSON-heavy, cloud-native, and that have dynamic web interfaces.

A few examples might be: - Workflow management - News aggregation - Todo apps and Calendars - Interactive forums / social products.



Break & Review



React.js

Declarative

React makes it painless to create interactive UIs. Design simple views for each state in your application, and React will efficiently update and render just the right components when your data changes.



React.js

Component-Based

Build encapsulated components that manage their own state, then compose them to make complex UIs.



React.js

Learn Once, Write Anywhere

React doesn't make assumptions about the rest of your technology stack, so you can develop new features in React without rewriting existing code.

React can also render on the server using Node and power mobile apps using React Native.



Node.js

As an asynchronous event-driven JavaScript runtime, Node.js is designed to build scalable network applications. In the following "hello world" example, many connections can be handled concurrently. Upon each connection, the callback is fired, but if there is no work to be done, Node.js will sleep.



```
const http = require('http');
```

```
const hostname = '127.0.0.1';
```

```
const port = 3000;
```

```
const server = http.createServer((req, res) => {
```

```
  res.statusCode = 200;
```

```
  res.setHeader('Content-Type', 'text/plain');
```

```
  res.end('Hello World');
```

```
});
```

```
server.listen(port, hostname, () => {
```

```
  console.log(`Server running at http://${hostname}:${port}/`);
```

```
});
```



Express.js

Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.



Mongodb Atlas

MongoDB Atlas is the global cloud database service for modern applications.

Deploy fully managed MongoDB across AWS, Google Cloud, and Azure with best-in-class automation and proven practices that guarantee availability, scalability, and compliance with the most demanding data security and privacy standards.



Exit Ticket

1. What is MERN
2. Define the components of MERN



References

MERN Stack - <https://www.mongodb.com/mern-stack>

Reactjs - <https://reactjs.org>

W3schools - <https://www.w3schools.com>

MDN Web Docs - <https://developer.mozilla.org>

Stackoverflow 2020 Developer Survey - <https://insights.stackoverflow.com/survey/2020>



Easter Eggs

Day Zero - <https://www.mongodb.com/day-zero?tck=brand-sitewidebanner>



Ask for Help

Phone Number - 233547557948

Skype username - pogakuofie

Email - mail@pogakuofie.com

Twitter - pogakuofie