

Comparative analysis of MEAN stack and MERN stack

¹Prati Jain, ²Manjunath C R

¹Student, ²Professor

¹Department of Computer Science,
¹Jain University, Bangalore, India.

Abstract: Current web applications are designed using a 'stack' of various technologies. A **stack** is a collection of frameworks and tools used to develop a software product. MEAN stack and MERN stack are two of the most popular and extremely powerful stacks used in web development. JavaScript provides an option to use both MEAN and MERN stack to develop web application by eliminating the need for switching the code. Both MEAN and MERN have a stellar combination of tools to help you build well-functioning software products. This ensures faster development of web applications and helps developers quickly get products to market as we only use a single language for frontend, backend rather than old ways of using different languages for frontend and backend and eventually leads to a huge cut in development costs and improves efficiency. Selection becomes intimidating for newcomers to pick the right stack for their applications. This paper aims at providing the various advantages and disadvantages of MEAN and MERN stacks, which enables the newcomers to pick appropriate right stack as applicable for developing a web application.

IndexTerms – MERN, MEAN, Angular, React.

I. INTRODUCTION

The need for technology has authorized the developers to work on both Front-end as well as on Back-end and much more. Full-stack developers tend to be skilled in the various streams which make up the domain of web development, viz. Frontend, Backend, Mobile app, and Testing. Earlier developers used to work on one or more of these technologies, but nowadays, the demand in the industry has raised and developers now have expertise in Frontend, Backend as well as ancillary technologies. This has led to the rise of the full stack developer.

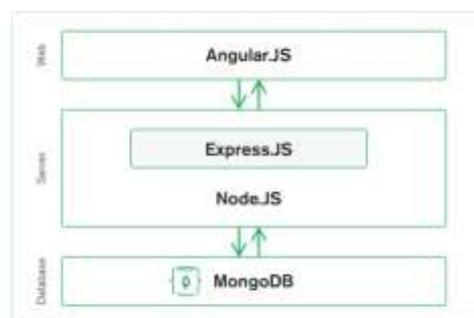
This means that the standard technique of mixing JavaScript, CSS, and hypertext markup language is no longer sufficient. For huge enterprise development of web applications using a stack has proven to be a sure shot solution to meet the demands of today's web audiences. JavaScript and its successors including JSX and ES6 enable the integration of multiple technologies all together. Nowadays, developers are able to use JavaScript in implementation of application logic, front-end experiences, and are also able to access databases.

Multiple stacks are used now days, but the dominant one's are as follows:

- MERN (MongoDB, Express.JS, React.JS, Node.JS)
- MEAN (MongoDB, Express.JS, Angular.JS, Node.JS)

II. MEAN STACK

MEAN stack is a collection of JavaScript technologies used in building complex websites or web apps. It is a full-stack JavaScript framework that proves to be helping hand in the application and web development, and also simplifies the work. The MEAN architecture is designed to make building web applications in JavaScript, and handling JSON, incredibly easy. Using JS makes a developer intensively comfortable as everything is performed using common concepts such as JS objects and asynchronous calls. The stack is made up of: AngularJS, ExpressJS, NodeJS, MongoDB.



A. ANGULAR.JS

Angular.js is a JavaScript-based open-source front-end web application framework maintained by Google and by a community of individuals and corporations to address most of the challenges faced in developing single-page applications. Angular.js allows you to extend your HTML tags with metadata in order to create dynamic, interactive web experiences much more powerfully than, say, building them yourself with static HTML and JavaScript (or jQuery).

Advantages of Angular

- 1) **Two-way data binding:** AngularJS was built with Model-View-Controller architecture. And the framework synchronized the Model and the View. As the data in the Model changes, the View does too. Two-way data binding allowed engineers to reduce development time as it didn't require writing additional code to provide continual View and Model synchronization
- 2) **REST friendly:** Representation State Transfer enables the application to rapidly interact with the server and fetch the data required to interact with the web pages.
- 3) **Enhanced Design Architecture:** Angular simplifies the way of managing these components even if a new programmer joins the project after the development process has already begun. The architecture is built in such a way that helps the programmer to **locate and develop** the code easily.
- 4) **Dependency Injection**
- 5) It supports the working of the **Model View Controller**.

Disadvantages of Angular

- 1) Angular is verbose and complex
- 2) Angular is gigantic. Mastering Angular over the very basic is definitely not a cakewalk.
- 3) UI lag issues on multiple user loads.
- 4) Dynamic applications didn't always perform that well. Complex SPAs could be laggy and inconvenient to use due to their size.

B. EXPRESS.JS

Express is a blazing fast, un-opinionated minimalist web framework for Node.js and is a web application framework for Node.js. It provides numerous features that make web application development super-fast and easy which otherwise takes more time using only Node.js. It's the back-end web application running on top of Node.js. Express.js has powerful models for URL routing and handling HTTP requests and responses. Express.js is designed using Node.js middle ware module called connect which implements http module. Henceforth most of middleware modules support unification with ExpressJS.

Advantages of Express

- 1) Makes Node.js web application development very fast and ultra-easy.
- 2) Easy to configure and customize.
- 3) You would be able to integrate several third-party applications and services with Express.JS.
- 4) Allows one to define routes of the application based on HTTP methods and URLs
- 5) Includes various middle ware modules which one can use to perform additional routines upon request and response.
- 6) Easy to connect with databases such as MongoDB, Redis, MySQL
- 7) Enables RESTful API server generation

Disadvantages of Express

- 1) Several client request problems are faced with the middleware system offered with Express.js.
- 2) There are several issues in the callbacks.

C. NODE.JS

Node.js is a JavaScript runtime environment that runs back-end application (via Express). NodeJS is based on Google's V8 JS JavaScript engine. To ensure outstanding performance, Node.js applies event-driven, non-blocking I/O paradigm. It is the runtime system that implements JavaScript on the back-end web application. It provides a number of features essential for developing web applications – including networking protocols such as HTTP. NPM is the default package manager, and allow developers access to tons of third-party modules in addition to the ones they create in their own application. Node is asynchronous, event driven, and non-blocking and multitasking. Node relies on callbacks to accomplish a task in succession after another task, but will continue executing the rest of the code that it can while multiple things are going on in the background.

Advantages of NodeJS

- 1) It is very fast
- 2) Open source and ever expanding.
- 3) Easier development process
- 4) Reusable code
- 5) Enhances overall productivity.

Disadvantages of NodeJS

- 1) Reduces performance when handling Heavy Computing Tasks
- 2) Node.js invites a lot of code changes due to Unstable API
- 3) Poor performance with relational databases.

D. MONGODB

MongoDB is an open-source, document database that provides persistence for app data and is designed with both scalability and developer agility in mind. MongoDB fixes the gap between key-value stores, which are fast and scalable, and relational databases, which have rich functionality. It is a type of NO-SQL database store where data is stored in form of key pair values instead of in a grid of rows and columns.

Advantages of MongoDB

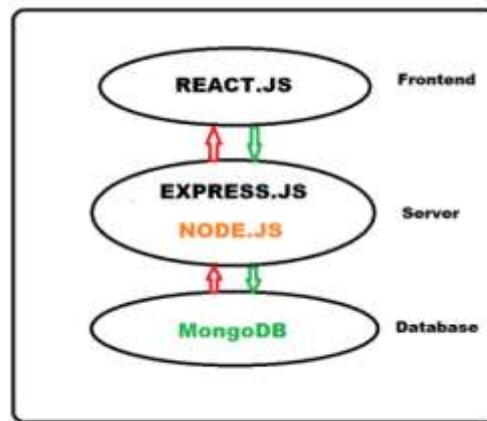
- 1) **Schema less** – MongoDB is a document database in which one collection holds different documents. Number of fields, content and size of the document can differ from one document to another.
- 2) **Structure of a single object is clear**
- 3) There are no complex joins in MongoDB.
- 4) Non requirement of conversion/mapping of application objects.
- 5) MongoDB provides the **facility of deep query** because it supports a powerful dynamic query on documents.

Disadvantages of MongoDB

- 1) Gigantic data size.
- 2) Less flexibility in running queries.

III. MERN STACK

MERN is a scaffolding tool which makes it easy to build universal apps using Mongo, Express, React and NodeJS. It minimizes the setup time and gets you up to speed using proven technologies. On comparing the two stacks, the flashing difference we see is the tech giant React.js replacing the former Angular.js. The stack is made up of: ReactJS, ExpressJS, NodeJS, MongoDB.



A. REACT.JS

React is a JavaScript library for building user interfaces. It is maintained by Facebook, Instagram and a community of individual developers and corporations. React enables development of large applications which supports the capability of loading new data without page refresh thus enabling speed of app and providing use a better surfing experience. This corresponds to View layer in the MVC pattern, and supports mingling with other JavaScript libraries and frameworks.

Advantages of ReactJs

- 1) Lightweight DOM for Better Performance
- 2) Virtual DOM support
- 3) Better performance
- 4) JSX support

Disadvantages of ReactJS

- 1) Covers only View layer in M-V-C - ReactJS Covers only the UI Layers of the app and nothing else. So, you still need to choose some other technologies to get a complete tooling set for development in the project.
- 2) Needs extra tooling and external support to make a web-application work

IV. ANGULAR VS REACT

While most developers can agree that the MEAN stack is a great option for modern app development, a large number are starting to advocate the use of React over Angular. Developers are now starting to talk in terms of MEAN vs. MERN, although it seems that the only difference being Angular or React. Angular is a front-end JavaScript framework whereas React is simply a JavaScript library. So, what is the difference? A framework is a structure for presenting code. It dictates a specific architecture for how the code is organized. Angular brings an M-V-C architecture to front-end development. It comes with additional helper functions and built-in functionality for making http requests, etc. Angular and React are both used for the same reason, to organize and render the presentation layer of the application.

1) Architecture

This goes back to the whole library vs. framework discussion. While React makes UI rendering a breeze, it's just a library. It's up to the developer, how one organizes his/her code to work with underlying data models, etc. As a framework, Angular enforces an

MVC like design, forcing developers to better organized their code. Although React is more flexible, it leaves more up to the developer as to how the app is organized. This can make the code harder to maintain. Besides, React offers only View layer, while Model and Controller are added with the usage of other libraries.

2) Performance

As shown in Table 1 React is arguably better for performance reasons. React's performance is greatly improved with the introduction of the virtual DOM. Since all virtual DOM trees are lightweight and built on server, the load on browser is reduced. While Angular 2 operates on real DOM. Imagine you want to update user profile details, let's say, their last name. Real DOM, instead of changing just that bit of info, updates the entire tree structure of HTML tables until it reaches the needed data. In our case, it's the last name. While virtual DOM allows us to update the changes without rewriting the entire HTML doc virtually. This renders updates much faster and ensures fast performance – no matter of the apps' size. Although Angular is a full front-end framework, the real DOM feature makes its apps slower when dealing with a ton of data requests. Although Angular 2 has introduced better state control, React is an easier and more intuitive way to handle change events.

3) Data Binding

AngularJS implements two-way data binding and a digest cycle to enable view layer synchronization with the underlying data layer. This proves to be extensively costly when hundreds of data items are needed to be updated dynamically. State change is detected using unidirectional data flow. This plays better with larger data sets, where hundreds of thousands of records need to be rendered and updated. React uses **one-way binding** when designing a React app developers often nest child components within higher-order parent components. The performance of Angular apps is negatively affected by bidirectional data-binding. One-way binding makes the code more stable, and also makes debugging an app build with React versus Angular app much easier. Still, the one/two-way binding of Angular is simpler to work with and makes the framework more flexible.

4) Third Party Libraries

React needs the support of a lot of third-party libraries in order to enable proper functioning of web-apps, like in the case of HTTP-Requests. React doesn't have an out of the box solution for making http calls to a backend server. Angular has a built-in http service wrapper that makes http requests a breeze. Although libraries like Axios allow one to easily make requests in React, it requires more configurations.

5) Popularity

Survey conducted through Stack Overflow it is observed **React.js** as the most preferred web framework by developers (**74.5%**) in 2020 whereas Angular observed **57.6%** of developers preferring to work with it.

Besides, here's what stateofjs.com survey (21,717 respondents) shows us:

React – 71.7% used React before, and WOULD use it again (**satisfaction**: 89.33% for 16,099 users)

Angular – 21.9% used Angular before, and WOULD use it again (**satisfaction**: 37.95% for 11,582 users)

According to Google Search Trends, during the initial years, Angular was in trend. But after year 2014, a huge alteration in trend is seen and even up to today, majority of people have shifter their interest towards development using ReactJS.

Table 1 Angular vs React

Parameter	Angular	React
Type	Framework	Library
Company	Google	Facebook
Language	TypeScript	JavaScript (JSX)
MVC	Present	View only
Performance	Better	Best
Third party support	Low	High
Size	746k	151k

V. CHOOSING THE RIGHT STACK

The selection of MEAN/MERN stack wholly depends upon the development and functional requirement of the system to be developed. Choosing either would provide you with a proficient software according to their capabilities. But for small to medium projects, performance measures of two stacks may not introduce a huge impact. Reason for that is developer's experience and comfort decide the speed of development and quality of end product so formed. If someone chooses a MEAN stack, the considerable overlap between the features in the two stacks play a crucial role in order to decide who does what within the development team. Henceforth one would have to decide where the core phase of the application's build will take place. As both AngularJS and ReactJS are equipped with overlapping features, either one can be used to parse the business logic of the web project. But traditionally ReactJS is chosen in industry because of the following reasons:

- Security to sensitive data.
- Extremely low latency.
- Actual code is hidden from user, making application extremely secure.
- Increased performance with the use of powerful servers.

Although, changes in trend may occur as now day's functionality is being transferred towards AngularJS as it runs on the user's browser. This helps in reducing overall memory requirement as well as improves the web response time. Besides this MEAN stack also has a reduced development cost. But there are other factors too that play a crucial role in the decision-making process. MERN is relatively faster when it comes to development speeds. Testing is also relatively easy with MERN because a lot of work has already been done with testing React.js so there are many common tools available that can be utilized. Table II shows various factors that must be considered before choosing any one of the two stacks. Thus, before any choice is made, all the factors must be carefully examined and tested according to the need of the project so that optimal use of features can be achieved.

Table 2 MEAN vs MERN

Parameter	MEAN	MERN
Tool	Low	High
Design	JS in HTML	JS only
JS Fatigue	Less	More
DOM	Regular	Virtual
Complexity	High	Low
Failure	Run time	Compile time

VI. CONCLUSION

It can be said that both MEAN and MERN stack are extremely reliable frameworks for blazing fast front-end development. However major difference lies in the way it is structured. This makes MEAN stack a better option for large-scale applications while MERN stack leads the race in the faster development of smaller applications. While React is definitively more powerful than Angular 1, Angular 2 accounts for the pitfalls seen with its predecessor. Both options play equally well with other stack components (MongoDB, Express, Node). Deciding which one is best for someone comes down to an emphasis on architecture, code maintainability, etc. One must remember that Angular 2 is the only true competitor to React.js when considering things like performance and server-side rendering.

REFERENCES

- [1] Pro MERN Stack Full Stack Web App Development with Mongo, Express, React, and Node by Vasan Subramaniam
- [2] Getting MEAN with Mongo, Express, Angular, and Node Book by Simon Holmes
- [3] Angular.io, 'AngularJS Documentation' [Online]. Available: <https://angular.io>. [Accessed: May- 2021]
- [4] MongoDB.com, 'MongoDB Official' [Online]. Available: <https://www.mongodb.com/>. [Accessed: May- 2021]
- [5] Expressjs.com, 'ExpressJS Official' [Online]. Available: <http://expressjs.com>. [Accessed: May- 2021]
- [6] ReactJS.org, 'ReactJS official' [Online]. Available: <http://reactjs.org>. [Accessed: May- 2021]
- [7] Nodejs.com, 'Nodejs Official' [Online]. Available: <https://nodejs.org>. [Accessed: May- 2021]