JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

# AN E-COMMERCE APPLICATION USING MEAN STACK

**[1]Abhijit Jain, [2]Vaibhav Ambhore, [3]Ajay kolekar, [4]Indrakumar Geete, [5]Prof. Rupali Parte**
[1,2,3,4] Students and [5]Assit. Prof. of Department of Computer Engineering,
Jayawantrao Saswant College of Engineering, Pune, India
Savitribai Phule Pune University, Pune
abhijitjain2949@gmail.com, vaibhavambhore1701@gmail.com, ajaykolekar879@gmail.com,
indrakumargeete15jan2001@gmail.com

***Abstract:*** This paper is based on the most advanced technology in the recent times that are employed to build a full stack e-commerce web application. The basic fundamental concepts of the MEAN stack are thoroughly explained along with the advanced features as well as their usage in the application as to how we can develop highly optimized website which can load quickly and ensure complete security from cyber threats. The MEAN stack is regarded as one of the most prevalent technology that is used in order to shape a full stack web application**.**

***IndexTerms*** **- node.js, express.js, mongodb, angular.js, server side, client side, etc.**

## I. INTRODUCTION

The MEAN is planned or created to build a powerful structure for helping programmers with utilizing better practices when working with the famous JavaScript components, assembling a robust system for supporting every day advancement needs, and settling normal issues with interfacing with MongoDB, Express.js, Node.js, Angular.js structures. For back-end and front-end, the MEAN stack is exceptionally basic and simple to utilize. There are a few innovations which utilize various dialects for both customer side and server-side execution. In MEAN innovation, we utilize just a single language for both customer and server-side. MongoDB offers a more adaptable, obliging layer for storing information. Node.js gives a superior nexus for running your server, while Express.js helps with normalizing how you build your websites. On the client server side, Angular.js gives a spotless method of adding intuitive capacities and AJAX-driven components.

## II. LITERATURE SURVEY

### Study of the Existing Models/Work

In the present day and age, innovation is developing at a fast speed. With the new innovations of equipment gadgets and frameworks, it is normal for software development innovations to progress too, effectively replacing the old innovations. Because of the expanding number of electronic gadgets that utilize the Internet and their real time nature of things, performance is critical. Generally, web improvement has been finished utilizing advances .For example, JAVA servlets, PHP (Hypertext Preprocessor), ASP.NET (Active Server Pages), ruby and so forth. While these advancements are extremely well known and have broad elements and long stretches of advancement, they have their own inadequacies contrasted with the present requirements with regards to performance. Node.js, Angular.js, MongoDB and numerous more have been as of late created to satisfy the present requirement for a superior other option. Google fostered the V8 JavaScript motor which gathers and executes the current JavaScript source code, oversees memory allotment and gives exact trash assortment which is pivotal for V8's superior presentation. NodeJS is a superior exhibition JavaScript runtime climate which is based on V8.

A famous improvement style that utilizes the V8 motor in a successful way is the MEAN stack. The MEAN stack enjoys the benefit of Node's bundle environment, npm (node bundle supervisor) which the biggest biological system of open source libraries. Hub uses JavaScript as its programming language for both server and customer side.

MongoDB is the data set used to store the information that application needs to run, Angular is the front-end application running on the customer side, Express is a lightweight HTTP (hypertext move convention) server system used to help the Angular application and the assets it needs to run the application, and Node.js is the climate used to run Express.

## III. EASE OF USE

Our application is very interactive and easy to use. We have created web server using express framework. In express we used JavaScript language so while sending response to client we don’t need to convert code because it’s already in JavaScript language. For front end we used Angular and our application is SPA. So it will be very faster and user will enjoy it. Last we deployed our application on AWS so we got all features of AWS.

## IV. RESEARCH METHODOLOGY

We have created two Node JS servers for admin and user portals respectively. Each routers are assigned to handle all requests form the server. We created modules for each database entries and instantiated in the server. We Imported Express module, then instantiated it and used as a middleware so all requests are processed through middleware, in the server itself.

For the front end part, we used Angular and created two servers for admin and user portal. Modules created in the main source folder, characterized by their properties and in each module we have created different components. Each components acts as a single page component and each components we have designed an HTML, CSS and typescript files. In order to connect Angular with the backend, we have used service files from Angular, so as to run this whole application on Node JS platform.

## THE MEAN STACK

Right now generally famous and broadly utilized open source web advancement stack is the LAMP (Linux, Apache, MySQL, and PHP) stack. Here Operating System utilized is Linux, Web server as Apache, MySQL as data set and PHP goes about as the programming language utilized for server side prearranging. A recently arising web improvement stack is the MEAN stack which utilizes MongoDB as information base, express as an adaptable waiter system that gives steering and handles solicitation and reaction, Angular works at the customer side.

### 3.1 Node.js

Node.js or just Node is the most important component of the MEAN stack. It provides the JavaScript development environment. It is built based on Google's V8 engine. Both Node and V8 are implemented in C and C++ for less memory consumption and faster performance. Node is based on Asynchronous I/O eventing model designed for developing scalable network applications. It fires callbacks on events, and each client event generates its own callback. If no work is to be done, Node is sleeping. While Node works on a single thread, it can serve many clients. Almost no function in Node directly performs I/O, they are handled by higher order functions. Node presents the event-loop as a runtime construct, but unlike some other technologies, node does not have a blocking start-the-event-loop call.

### 3.2 Express.js

Express is a server-side framework built in the Node.js environment. It handles the client requests to the server and manages routing and HTTP methods such as GET, POST, PUT etc. By calling the express() exported by express e.g. app = express(), an express application is created. The app object is used to perform various operations and provide services by express. Express listens to a socket connection on a path or on a specified host and port number. Then using one of the METHOD() functions such as app. get() where app is the express application object and get() is the METHOD function, start the request-response-cycle of the appropriate middleware. To configure middle wares the app. Route() returns an instance of a single route, which can be handles by HTTP methods and optionally middle wares. The app, render() is used to render HTML view files using a call back. Express uses template view engines to render views.

### 3.3 Angular.js

It is an open source JavaScript library developed and maintained by Google. It was developed with capabilities to handle the entire client side application and interaction. It's specifically used to develop a SPA (Single Page Application) that loads the whole web page on an initial request. Angular has the ability to perform client side routing. Another specialty of Angular is that it is a MVW (Model View Whatever) architecture i.e. the developers have the freedom to choose whichever way they want to implement Angular for their projects. Angular uses Directives, which are HTML mark-ups which appear as html elements, attributes or even CSS classes. The directives are used to bind data, and DOM manipulations. The directive ng-app is used to define the Angular application. Views and models are managed by a controller. Angular can handle the entire client-side routing. This is done using the directive Routes. We can call controllers using the routes, and similarly render templates when necessary. The SPA ability of Angular is achieved through routing.

### 3.4 MongoDB

MongoDb is a document oriented NoSQL type of database. It stores data in JSON format. It has a dynamic schema and hence very popularly used to develop scalable applications. MongoDb does not require its users to know a traditional relational language such as SQL. Node has an inbuilt package called mongoose that maintains the complete interaction between MongoDb and the node, objects or documents is the most common form of data storage in. Binary encoded JSON also known as BSON is the document format. BSON is extended from JSON and has a few extra data types that are not supported in JSON. Mongoose is used to perform CRUD (Create Read Update Delete) operations on MongoDB.

## V. PROPOSED SYSTEM

The MEAN.js is designed or developed to make a vigorous framework that allows developers use efficient practices while they are working with the popular JavaScript components, which in turn helps in supporting usual development needs, and solving common issues by connecting to MongoDB, Express.js, Node.js, AngularJS frameworks. The whole application is going to be built on a NodeJS platform which is the runtime environment of JavaScript. Old applications were used to build on the different platforms which were used to be run by their own compilers like java, ruby c#, python etc. these were the languages used at backend which slowdowns the performance as compared to the MEAN, because of absence of rendering JavaScript engine.

Now JavaScript introduced at the backend too which is asynchronous, lightweight, and faster as compare to other programming languages in web applications.

- Angular.js is a client-side language which is written in JavaScript. So, at first the client request is processed.
- After that, the client request then enters the server (Node.js). Node.js acts as a server-side language written in JavaScript.
- Express.js then makes the request to the database.
- After receiving a request, MongoDB then retrieves the data and returns the response back to the Express.js.
- The response from Express.js is sent back to the Node.js, after which it is forwarded to the Angular.js by Node.js for displaying the final result.

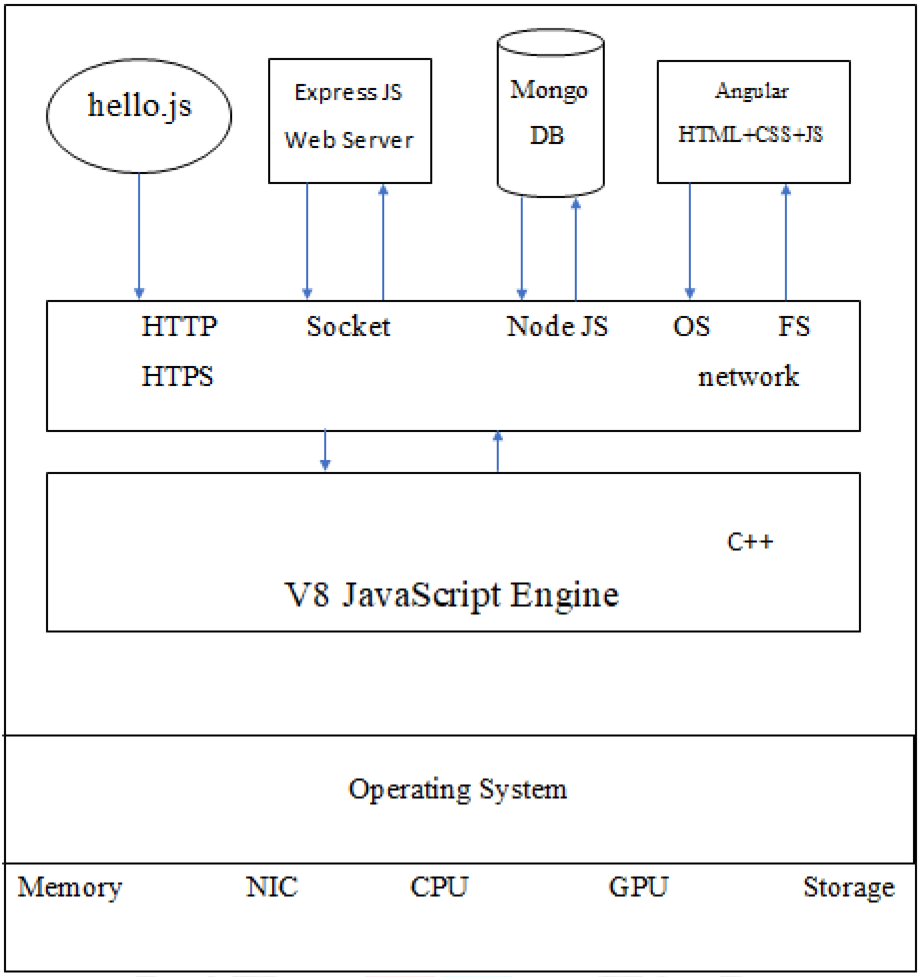

Fig. 1: Architecture of MEAN stack

### Architecture Description (Node.js)

Node.js works on a single thread, utilizing non-blocking I/O calls, permitting it to help a huge number of concurrent connections without causing the expense of thread context switching. The design of dividing a single thread among every one of the requests that use the observer pattern is expected for building exceptionally simultaneous applications, where any function performing I/O should utilize a callback. To oblige the single-strung occasion circle, Node.js uses the libuv library which thus utilizes a fixed-sized thread pool that is responsible for a portion of the non-blocking I/O operations.

### Architecture Description (MongoDB)

In MongoDB, data is represented as a collection of JSON documents.

MongoDB's querying process is object-oriented, which basically means we can pass MongoDB a document explaining whatever needs to be queried. Joints are not supported by MongoDB; however, it usually supports multi-dimensional data types, which includes documents and arrays.

MongoDB; will only have an collection of different comments and a collection of posts present within a post. One of the noteworthy features of MongoDB is that, we don't have to define the schema for the database.

### Architecture Description (Express.js)

After the user procedures and sends a request via Angular.js then that request is at first retrieved by the Node.js. The Express.js then hosts the website that is shaped for the Node.js. Node.js and Express.js are both server-side languages. After CRUD operations, the API data is finally retrieved from the MongoDB and then finally send to the User.

## VI. CONCLUSION

Mean stack technology is improving day by day with various updates and improved functionalities and is also easy to use and convenient. The MEAN stack not only ensures a robust and maintainable solution for creating an ecommerce web application but also helps to keep the whole process secure by using JSON Web token, which is a standard that defines a condensed and self-sufficient method to steadily transfer data between two or multiple parties as JSON object. This web application has a lot of potential to scale the application on a larger platform as is evident from the AWS EC2 instance created which allows the application to be deployed globally for easy access through the World Wide Web. Furthermore, e-commerce applications are not usually built with any other existing stacks like MERN or Java application based. So using MEAN stack gives us an edge over other existing applications, mainly in terms of high computation potential, high performance benchmark and being lightweight.

## VII. ACKNOWLEDGMENT

We would prefer to give thanks the researchers likewise publishers for creating their resources available. I'm conjointly grateful to guide, reviewer for their valuable suggestions and also thank the college authorities for providing the required infrastructure and support.

## REFERENCES

[1] J. Tihomirovs and J. Grabis, "Comparison of SOAP and REST Based Web Services Using Software Evaluation Metrics," Inf. Technol. Manag. Sci., vol. 19, no. 1, pp. 92– 97, 2016.

[2] R. T. Fielding and R. N. Taylor, "Principled design of the modern Web architecture," ACM Trans. Internet Technol., vol. 2, no. 2, pp. 115–150, 2002.

[3] S. Mumbaikar and P. Padiya, "Web Services Based On SOAP and REST Principles," Int. J. Sci. Res. Publ., vol. 3, no. 5, pp. 3–6, 2013.

[4] M. Stajcer and D. Orescanin, "Using MEAN stack for development of GUI in realtime big data architecture," 2016 39th Int. Conv. Inf. Commun. Technol. Electron. Microelectron.

[5] RIPRO 2017 - Proc., pp. 534–529, 2016.

[6] J. Poulter, S. J. Johnston, and S. J. Cox, "Using the MEAN stack to implement a RESTful service for an Internet of Things application," IEEE World Forum Internet Things, WF-IoT 2015 - Proc., pp. 280–285, 2015.

[7] F. Bocchio, "Estudio Comparativo De Plataformas Cloud Computing Para Arquitecturas Soa," vol. 1, no. 5, p. 114, 2013.