

Web Frameworks, Web Stacks and Databases

Mishra Rishikesh¹, Nikhil Tiwari², Aditya Shrivastava³, Dr. Pallavi Goel⁴

^{1, 2, 3, 4}Dept of Computer Science

^{1, 2, 3, 4}Galgotias University

Abstract- Online applications have become a very important component in today's world. Web apps have helped businesses to grow and has eased the requirements and have helped the business to achieve their goals in pace. There are different stages/levels involved in development of a web application, for example requirement analysis, designing stage, implementation, testing and maintenance. Frameworks are like classes that provide pre-built functionalities and methods. Applications are built using these frameworks because they already provide fundamental requirements which are required in development of any web application. There are many different web development frameworks available for developing both front-end back-end of a web application with support for different programming languages. Database in web development is required as a place where the web application stores its data and is used by back-end frameworks. The term Web stack means a combination of web frameworks and databases. Selecting the correct Web Stack for development is a crucial part in a web development lifecycle.

Keywords- back-end, databases, development, front-end, requirement analysis, designing, web frameworks, web stack.

I. INTRODUCTION

In today's world, advanced with the internet there are too many technologies available for developing a web application, selecting the right fusion of technologies available for developing a web application is a very decisive task. A web application can be categorized into two parts namely front-end and back-end. These two services communicate with each other with the help of API calls. The front-end of an application is responsible for handling the UI while the back-end of an application is responsible for handling the actual logic of an application. Having two separate services handling different parts separately gives a lot of advantages like quick and easy development, upgradation, and maintainability. But using two separate services can have some problems too. This paper aims to discuss various frameworks, and solution stacks, available for developing a dynamic and modern web application. The paper is divided into different sections focusing on different concepts and the final section concludes the paper.

FRAMEWORKS, DATABASE & STACKS

1. Front-End Web Frameworks

The component of a web application that an end-user view or interacts with is called the Front-End of the application. It includes everything like images, colors, text, various graphics, buttons and other similar components, etc. HTML is used for designing the structure of web page and CSS is used for styling the page, Validations and logics are managed with the help of JavaScript. The front-end developers need to make sure that the website has a great performance and is responsive and it displays accordingly with different devices of different sizes and scales. None of the components of a web application should respond irregularly.

1.1 ReactJS

ReactJS is an open-source JavaScript library used as a base for developing UI components for single-page or mobile applications. It is developed and maintained by Facebook and a community of developers. React involves state management and rendering that state to the Document Object Model (DOM), thus additional libraries for routing are usually required with React applications. "Components" are small segments of code that is used to build complex UIs. The components are also reusable which makes the development easier. Writing HTML inside JavaScript is simplified with using JSX. JSX is an extension to JavaScript language.

1.2 AngularJS

AngularJS is a JavaScript-based open-source front-end framework to resolve the hurdles encountered in development of single-page applications. AngularJS is developed and maintained by Google and a community. Angular provides a framework for server-side. HTML is used as template language in AngularJS, it reads the HTML page, which consists of custom HTML attributes embedded into it. The dependency injection and data binding features of Angular eliminates the necessity of extra code and this process happens within the browser, which makes AngularJS compatible with any server technology.

1.3 VueJS

VueJS is a very well-known and an open-source framework used for developing UIs and single page applications. VueJS focuses on the view layer of the application and component composition and is designed to be incrementally adaptable. VueJS is developed and maintained by Evan You and the core team. Vue uses HTML attributes in form of "directives".

Table -1: Front-endFrameworks Comparison

Comparison Parameter	ReactJS	AngularJS
Packaging	Strong	Weak
Rendering	Client Side	Server Side
Founders	Maintained by Facebook	Maintained by Google
Architecture	Virtual DOM	MVC (Model View Controller)
Written in	JavaScript	TypeScript
Data Binding	One Way	Two Way
Learning curve	Low	High

II. BACK-END FRAMEWORKS

Back-end consists of countless functions. For example, protecting APIs against external threats and attacks, to validate users, etc. the back-end frameworks allow developers to protect against these threats and provides functionality to the web application.

These frameworks are selected with respect to different factors. For e.g., programming methods, interfaces they support, language that it uses. In addition to all this, frameworks provide templates and various tools to track websites development activities. Using a Back-end framework increases the pace of development and thus saves time. These frameworks are very efficient when it comes to scaling. Some very popular and important frameworks are discussed in the sub- section.

2.1 Django

Django an open-source web framework. It’s very quick because it is based on python and is designed for quick conversion of any concept into a web application. It allows the user to focus on developing while it handles other complex tasks by itself unlike the traditional “struggle web development”. Django follows model-template-views that takes care of most of the required services of web developments like user authentication, content management etc. by itself. Django is also being used in many big scale and popular websites like Instagram and Pinterest. The controller is responsible for passing data from model to view.

2.2 NodeJS

NodeJS is an open-source, JavaScript runtime environment which functions on the V8 engine by chrome and helps to execute JavaScript code outside a web browser. NodeJS is designed to manage multiple communications. Call back is triggered only when required and is available for every individual connection. This behavior is not found in any other models, where the operating system threads are used. There are no locks in NodeJS because there is no function present that performs I/O, so the process never stops and thus deadlock is never formed which makes NodeJS scalable. NPM (Node Package Manager) which has bunch of open-source libraries for development and other various purposes is a NodeJS Package ecosystem.

2.3 Spring Boot

Spring boot is extension of spring that is used for developing spring-based stand-alone applications. Spring Boot aims at development of production-ready applications. The spring boot adheres well with microservices but is not recommended for big applications. The feature of directly deploying it into docker containers reduces the complexities and makes developing web applications easy. Spring Boot provides a development model which works well with developing third-party UI and REST APIs. Spring Boot requires no code generation and requirement for XML configuration. Many authentication agreements like SAML, OAuth and LDAP are supported by Spring Boot. It also supports many different databases either it be relational and non-relational, cloud-based services, and frameworks like Spark, Flume, and programming languages like Java and Kotlin. Spring Boot has the disadvantage of longer development time.

Table – 2: Back-end frameworks comparison

Comparison Parameter	Django	NodeJS
Costs	Open-Source	Open-Source
Based	Python	JavaScript
Scalability	Less scalable	More scalable comparatively
Architecture	Model-Template-View	Event-driven
Performance	Good	Better
Complexity	More than NodeJS	Less
Security	Most of the basic security concerns are handled by Django.	Developers need to make sure of the security.

III. DATABASES

Selecting a database is based on selecting between relational (SQL) and non-relational (NoSQL) databases. Both types of databases work successfully but the user needs to remember that there are some pros and cons of every database. Each database aims at specific purpose and is not able to meet all the requirements. Thus, selection of correct database is very crucial and should be done after analyzing application needs and aspects. The requirements of small-scale data and storing OLTP data can be fulfilled by relational databases. A fixed schema cannot be focused because of unstructured data and thus, non-relational databases are preferred for them.

3.1 PostgreSQL

PostgreSQL is well suited for OLTP kind of workloads as it is compatible with ACID properties. PostgreSQL is an analytical database. PostgreSQL is very much compatible with mathematical softwares. PostgreSQL is scalable and it is well suited for modern websites that have very high traffic. PostgreSQL is integrable with modern frameworks and thus allows the application to meet thousands of requests every second. Scaling up to many database servers is very easy with PostgreSQL because of its excellent replication features. PostgreSQL can be used in applications that supports both relational and non-relational databases.

3.2 MongoDB

MongoDB is a non-relational database which does not use relational databases components like tables and rows, but it uses objects and is document-oriented. Data is present in key-value pairs which make it the fundamental unit in MongoDB and grouping that data is known as collections. MongoDB is scalable and has capability to replicate. JSON documents are represented using BSON format in MongoDB. BSON makes the process of encoding and decoding efficient. In order to resolve the network partitions MongoDB maintains consistency but also compromises with the availability.

Table – 3: Databases Comparison

Comparison Parameter	PostgreSQL	MongoDB
Architecture	Monolithic	Distributed
Platform	Unix-like and windows	More compared to PostgreSQL
Scaling	No built-in mechanism for horizontal scaling	Sharding is used for horizontal scaling.
Validation	Does not have in-built mechanism for schema validation	Supports schema validation through the IETF standard
Schema	Strongly typed	Flexible
Transaction	ACID	Multi-document ACID Transactions
Orientation	Document Oriented	Object Oriented

IV. WEB STACKS

A web stack is a combination of framework and database that are required for developing web applications and websites. A web stack is a set of solution used for performing specific tasks. Web stacks are important for web applications and also for the websites. A web stack should consist of an OS, a database software, any programming language and even a web server, all this at least at the minimum level.

4.1 MEAN/MERN/MEVN

Mongo, ExpressJS, AngularJS/ReactJS/VueJS, NodeJS

This web stack is used for building dynamic websites and applications. This stack supports developing both the front-end and the back-end in a single language i.e., JavaScript which can be considered as the main advantage of this stack. MongoDB, which is a NoSQL database, is used for data storage. ExpressJS and NodeJS are used for backend. React, Angular and Vue are used for developing the front-end of the application.

4.2 SPRING STACK

Spring stack allows building dynamic web solutions. Spring Boot offers prebuilt tools to pace development reducing the need for configuring application servers. Building REST Api services using spring boot is simple and not very time consuming. With the usage of JPA and Hibernate database connectivity is greatly simplified. Spring

Boot also has features to provide an embedded application server.

4.3 LAMP/LEMP

Linux, Apache/Nginx, MySQL, PHP

A very popular web stack with involvement of an operating system i.e., Linux, a web server, a relational database software and a scripting language. The main aim of this stack is to develop dynamic web sites and applications. This stack is a type of layered structure where each layer is built on top of the previous layer. Operating system is the base layer; the web server is on top of the OS. The Database layer stores and manipulates the data and the scripting language is used to make the logic and display it.

V. CONCLUSION

Every web framework, database and web stack have its pros and cons with it. Various crucial factors like time required, scalability and maintenance, programming language, developer's proficiency with the technologies and others affect the selection of appropriate web stack. Database is also chosen after considering the type of data, availability and consistency required by the application. Thus the requirements are of importance and taken into consideration for selecting web framework, database and web stack.

REFERENCES

- [1] Reactjs.org. Available at:
<https://reactjs.org/tutorial/tutorial.html>
- [2] Angular.io. Available at: <https://angular.io/docs>
- [3] <https://en.wikipedia.org/wiki/AngularJS>
- [4] [https://en.wikipedia.org/wiki/React_\(JavaScript_library\)](https://en.wikipedia.org/wiki/React_(JavaScript_library))
- [5] <https://en.wikipedia.org/wiki/Vue.js>
- [6] [https://en.wikipedia.org/wiki/Django_\(web_framework\)](https://en.wikipedia.org/wiki/Django_(web_framework))
- [7] Nodejs.org Available at <https://nodejs.org/>
- [8] Docs.djangoproject.com.
<https://docs.djangoproject.com/en/3.0/>
- [9] https://en.wikipedia.org/wiki/Spring_Framework#Spring_Boot
- [10] Mongoddb.com
www.mongoddb.com/compare/mongoddbpostgresql
- [11] www.techopedia.com. Web Stack.