Contents

UNIT 9 – INTRODUCTION TO MERN

9.0 Structure of Introduction to MERN

9.1 Learning Outcomes:

After the successful completion of this unit, the student will be able to:

- Explain the concept of Full-stack development[L2]
- Discuss the concept and benefits of MERN[L2]
- Explain the components of the MERN.[L2]
- Discuss the tools and libraries used with MERN.[L2]

9.2 Overview of Full Stack Development

Multiple technologies are used for developing any web application. These technologies together form a “stack”. A technology stack refers to a collection of frameworks, tools, and libraries that are carefully selected and combined to work together in the development of a complex and functional software product. The most popular technology stack is the “LAMP” stack which uses all open-source software and is an abbreviation for Linux, Apache, MySQL, and PHP. Nowadays, web applications that use the Single Page Application(SPA) concept are becoming more popular. They use lightweight server calls to retrieve data from the server and make changes only to some parts of the web pages without having to reload

the entire page. This has brought rise to front-end and back-end frameworks for developing web applications. Full-stack development technology is one of them.

Full stack development is the ability to develop web-based applications using both front-end skills and back-end skills. The front end refers to the client software and consists of the user interface that users see and interact with whereas the back end refers to the server software that takes care of the business logic that connects the application to the other services and the database. The back end of an application is responsible for data storage and retrieval and also for processing the requests sent to the server.

The front end uses technologies like HTML, CSS, and JavaScript. JavaScript is widely used in front-end development but recently it is also used for back-end development. To create reusable components and manage the states of the applications front-end frameworks like React and Angular are most commonly used.

The back end uses server-side programming languages like Python, Node.js, Ruby on Rails, etc. To simplify the development process the most commonly used back-end frameworks are Express.js, Flask, and Ruby. For managing and storing data the common databases used are MongoDB, MySQL, PostgreSQL, Firebase, etc.

Overall, full-stack development uses various technology stacks for application development that include a wide range of technologies and components which can vary depending on the specific requirements of the application.

Some of the commonly used stacks in full-stack development are

- MERN – [MongoDB, ExpressJS, ReactJS, NodeJS]
- MEAN – [MongoDB, ExpressJS, AngularJS, NodeJS]

Full-stack developers need to have a broad range of technical skills and be able to work collaboratively with designers, project managers, and other developers to create a functional and user-friendly application.

9.3 Overview of MERN

MERN is a widely used abbreviation that stands for MongoDB, Express.js, React.js, and Node.js.

1. M in MERN stands for MongoDB. It is a type of database that is popular among developers. It's called a NoSQL (Non-Structured Query Language) database because it doesn't use a traditional query language.

2. E in MERN stands for ExpressJS. It is a powerful tool that acts as a middleware and works with Node.js which is another technology in the MERN stack. It is used to help developers create web applications.
3. R in MERN stands for ReactJS. It is a client-side JavaScript library that is used to create rich and interactive user interfaces for web applications.
4. N in MERN stands for NodeJS. It is a runtime environment that is used to run JavaScript. Node.js is fast and efficient in code execution.

MERN is a combination of technologies that are commonly used for developing full-stack web applications. Together, these technologies make it easier for developers to create web applications that are fast, responsive, and easy to use. Many developers like to use MERN because it is a flexible and scalable option for building web applications.

9.3.1 MERN Architecture:

The MERN stack is typically used to implement a three-tier architecture in web development, also known as a client-server model as shown in Figure 9.1.

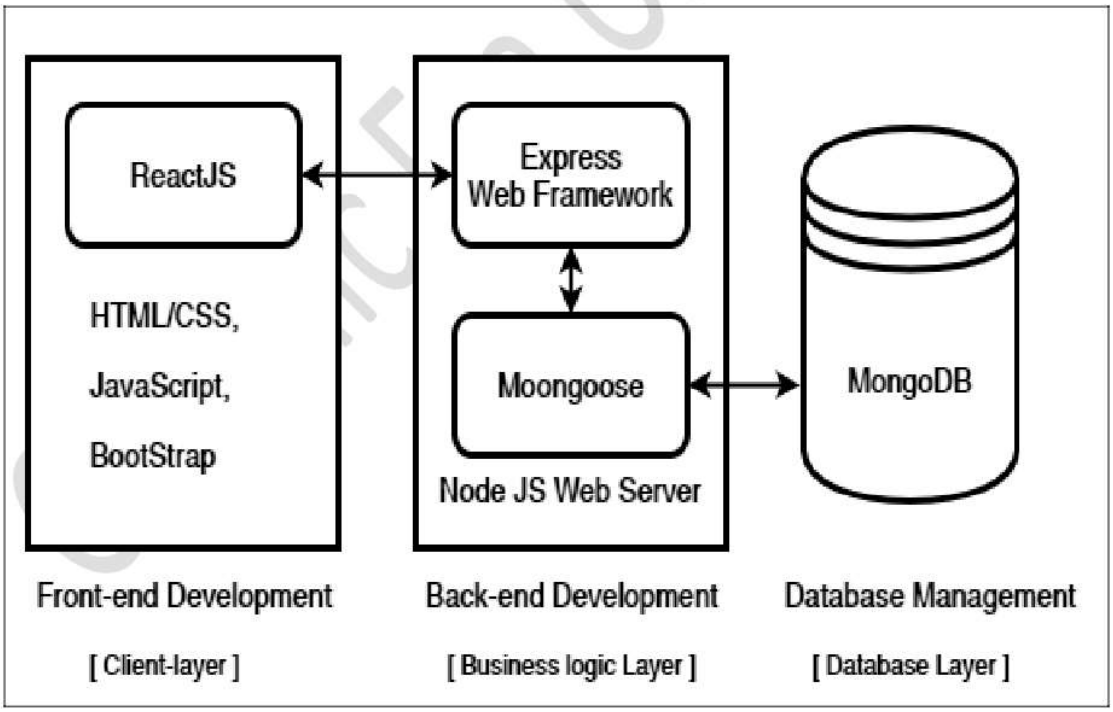

Figure 9.1: MERN Stack Architecture

The first tier is the topmost layer of the MERN stack that is handled by ReactJS. ReactJS is an important component of the MERN stack that is used to develop the client side of web

applications where the user interface and related functionality run in the user's web browser. It is known for creating dynamic client-side applications and is commonly used to create complex interfaces. Using ReactJS saves time as it allows the reusability of code. It also helps to change some parts of the web page without having to reload the entire page.
The second tier is the second level from the top layer and is handled by Express.js and Node.js, which are two components of the stack. Express and Node.js work together in the MERN stack to help developers create the server side of web applications. This layer is the server side, where the application's business logic, handling of requests from the client side, and interaction with the database occurs.
The third tier is the database layer, where MongoDB is used as the database system in the MERN stack. This layer is responsible for storing and retrieving data from the database, which is critical for many web applications. It is the most popular NoSQL database.

9.3.2 Benefits of MERN Stack:

The MERN stack offers several benefits for web and mobile application development:

- Open-source: MERN's open-source components provide access and support to a large development community.
- Cost-effective: All the components of the MERN stack - MongoDB, Express.js, React, and Node.js - are open source and built using JavaScript, making it cost-effective and delivering excellent results.
- Faster delivery: MERN-based web and mobile applications are designed in such a way that can be delivered quickly to clients.
- Better performance: The MERN stack has better performance because of its fast response times between the front-end, back-end, and database, resulting in a smoother user experience.
- Improved security: The stack includes security features that help safeguard web applications and APIs from potential threats.
- Faster modifications: Quick modifications to client requests are supported by MERN technologies, making it ideal for agile development.
- Search Engine Optimization (SEO) - friendly: Websites created with MERN technologies, allow search engines like Google, Yahoo, and others to efficiently index the website's pages and are designed to be search engine optimization (SEO)-friendly.
- Easy switching: The simplicity of MERN's single language approach allows for easy switching between client and server, further streamlining the development process.

9.4 MERN Components

The main components of MERN are

- ReactJS
- Node.js
- ExpressJS
- MongoDB

9.4.1 MongoDB

- MongoDB is one of the most popular open-source document-oriented NoSQL databases that is used to store, retrieve, and manage large amounts of data.
- It doesn't rely on a fixed schema. This means data is stored differently compared to traditional databases. Instead of using tables and rows, MongoDB stores data in a separate document. This makes it easier for developers to create flexible data structures that can be easily parsed, which in turn improves speed and scalability.
- MongoDB is also faster than traditional databases because of its efficient storage and indexing techniques.
- Additionally, MongoDB supports a flexible document model that's easy for developers to work with.
- It's also easy to scale MongoDB by adding more servers, which can increase productivity for developers working with large datasets.
- It is commonly used in web applications, mobile applications, and other types of software that require a scalable, high-performance database solution
- Query language used in MongoDB is based on JSON.
- In MongoDB, JSON format is mainly used for data interchange.

9.4.2 Express

- Express is a server-side framework for Node.js that's based on JavaScript. As one of the best backend development frameworks, it provides developers with a powerful platform to create and maintain robust servers. It's most commonly used for building web and mobile applications efficiently and straightforwardly.
- Express makes it simple and easy for developers to create robust APIs and web servers with simple syntax.
- Additionally, Express makes it easy to connect with databases like MongoDB and handle errors using error-handling middleware.
- With its extensive collection of third-party add-ons, Express provides better functionality, improves software speed, and enhances security.

- The built-in router in Express promotes code reusability and makes it easier for developers to create reliable and scalable applications.

9.4.3 ReactJS

- React is a widely used open-source JavaScript library that's designed for developing web applications. Specifically, it's intended to create user interfaces for single-page web applications. React is also known as ReactJS or React.js.
- Developed by Facebook, it's important to note that React is not a JavaScript framework, but rather a JavaScript library.
- One of the key benefits of using React is that it enables developers to create UI components that are reusable, fast, simple, and scalable.
- React views are declarative, the developer specifies what they want the program to do, and the framework or library they are using figures out how to do it. In other words, the developer describes the desired result, and the framework takes care of the implementation details.
- React is a component-based library, which means that UI elements in React are created as individual, self-contained components. Components in React are like building blocks that can be composed together to create complex UIs. Each component can have its state and props and can be reused throughout an application.
- Additionally, React can be used in combination with other JavaScript libraries or frameworks. The performance of React is impressive, the immutability of data helps ensure very fast data processing.

9.4.4 Node.js

- Node.js is an open-source environment used for server-side programming. It's a cross-platform runtime environment that enables JavaScript code to be executed outside of a web browser. Although it's often used for building real-time web applications due to its fast synchronization, it's important to note that Node.js is not a programming language or a framework. Node.js runtime runs JavaScript programs similar to Java Runtime which runs Java programs.
- It's frequently employed for developing numerous backend services for web and mobile applications.
- The event loop feature in Node.js enables it to have a non-blocking and asynchronous nature, which leads to quicker operation processing. An event loop is a mechanism that enables Node.js to handle multiple events/requests concurrently without blocking the main thread.

- Additionally, Node.js offers caching properties that make it easy to cache a single module so that lines of code do not have to be re-executed, resulting in faster response times.
- Node.js is highly consistent and provides extremely robust and quick services.

9.5 Tools and Libraries

The MERN stack is a widely used web development technology stack consisting of four main technologies: MongoDB, Express.js, React.js, and Node.js. To make development with MERN more efficient and effective, developers often utilize various tools and libraries specifically designed for each of these components. Some commonly used libraries are :

React Router: a routing library that allows you to easily handle navigation and transition without loading the entire page and routing within the React.js application.

React Bootstrap: is a popular open-source library that provides a set of pre-designed UI components for React applications, using the Bootstrap design system. Bootstrap is a widely used front-end framework for building responsive, mobile-first web applications.

React-Bootstrap allows developers to easily create professional-looking UIs by providing a set of ready-to-use components such as buttons, forms, modals, and navigation menus. These components are built using React, and they are fully customizable and extensible.

Webpack: is a powerful tool that can help developers build efficient, modular JavaScript applications that can be deployed to a variety of environments.

Apart from these, some examples of commonly used tools and libraries that developers use to work with MERN are:

1. MongoDB tools and libraries-

   MongoDB Compass: a graphical user interface for MongoDB that allows developers to visually explore and manipulate their data.

   MongoDB Connector for BI: a tool that allows developers to connect their MongoDB data to popular business intelligence (BI) tools such as Tableau, Power BI, and QlikView.

   Mongoose: a library that simplifies interactions with MongoDB by providing a more expressive API for data modeling and validation.

   Robo 3T: a lightweight MongoDB management tool with a clean and intuitive interface.

   PyMongo: a Python driver for MongoDB that allows developers to interact with MongoDB from their Python applications.

   MongoDB Atlas: a cloud-based database service that provides a fully managed MongoDB deployment with automatic scaling, backup, and recovery.

MongoDB Shell: a command-line interface for interacting with MongoDB, allowing developers to run queries, insert and modify data, and manage the database.

GridFS: a specification for storing and retrieving large files such as images, videos, and audio files in MongoDB.

Mongo-express: a web-based admin interface for MongoDB that allows users to manage databases, collections, and documents through a simple and intuitive interface.

2. ExpressJS tools and libraries-

Body-parser: a middleware that parses incoming request bodies in a middleware before the handlers, making it easy to handle form data, JSON, and other types of requests.

Morgan: a middleware that logs HTTP requests to the console, making it easy to debug and monitor the application.

Passport: a middleware that provides authentication for Express.js applications, supporting a wide range of authentication strategies.

Express-validator: a middleware that provides validation and sanitization of user input, ensuring that the application is secure and free of vulnerabilities.

Helmet: a middleware that adds an extra layer of security to the Express.js application, by setting various HTTP headers to protect against common web vulnerabilities.

Compression: a middleware that compresses HTTP responses, reducing the size of data sent to clients and improving application performance.

Connect-flash: a middleware that allows storing messages in the session and retrieving them later for display on the website.

Express-session: a middleware that provides session management for Express.js applications, allowing to store the user data across multiple requests.

Multer: a middleware that provides file uploading capabilities for the Express.js application, making it easy to handle file uploads from forms.

3. ReactJS tools and libraries-

Redux: a predictable state container that allows management of the state of the application predictably.

React Router: a routing library that allows you to easily handle navigation and routing within the React.js application.

React-Redux: a library that integrates the Redux state container with React.js, making it easy to manage the state of the React.js application using Redux.

Axios: a promise-based HTTP client that makes it easy to send asynchronous HTTP requests from the React.js application.

React Bootstrap: a set of reusable UI components that are built on top of the Bootstrap framework, making it easy to create responsive and mobile-first user interfaces.

Styled Components: a library that allows to write CSS in the JavaScript code, making it easy to create reusable and composable styles for the React.js components.

React Native: a framework that allows, the building of mobile applications for iOS and Android using React.js and JavaScript.

Enzyme: a testing utility for React.js that makes it easy to test the React.js components in isolation.

Material-UI: a set of React.js components that are designed according to Google's Material Design guidelines, making it easy to create beautiful and responsive user interfaces.

Jest: is used to test React applications.

These are just a few examples of the many tools and libraries available for React.js. The choice of which ones to use will depend on the specific requirements of the project.

4. NodeJS tools and libraries

NPM (Node Package Manager): It's a package manager for Node.js that allows to install and manage dependencies for Node.js applications.

Express: It's a web application framework for Node.js that provides a robust set of features for web and mobile applications.

Mocha: It's a testing framework for Node.js that allows, to write unit and integration tests for the Node.js applications.

PM2: It's a process manager for Node.js applications that makes it easy to manage and scale the Node.js applications.

Socket.io: It's a library that allows, implementing real-time communication between clients and servers in Node.js applications.

Passport: It's a library that provides authentication for Node.js applications, supporting a wide range of authentication strategies.

Sequelize: It's an ORM (Object-Relational Mapping) library for Node.js that allows interaction with relational databases such as MySQL, PostgreSQL, and SQLite.

Winston: It's a logging library for Node.js that allows logging messages and errors in the Node.js applications.

Sharp: It's a library for Node.js that allows manipulation of images, such as resizing, cropping, and compressing, making it easy to implement image processing in Node.js applications.

These are just a few examples of the many tools and libraries available for Node.js. The choice of which ones to use will depend on the specific requirements of the project.

9.6 React Library

React plays a critical role in the MERN stack, serving as the library for building user interfaces on the client side. In the MERN stack, React is used to create reusable, component-based UIs that can be rendered in the browser.

Reacts declarative and component-based approach makes it easy to create complex UIs by breaking them down into smaller, reusable components. This can lead to faster development times and improved code maintainability.

Reacts virtual DOM also plays a key role in the MERN stack's performance. By only updating the parts of the UI that have changed, React can minimize the amount of work required to update the UI, leading to faster rendering times and improved performance.

Additionally, Reacts ability to easily integrate with other libraries and tools in the MERN stack (such as Redux for managing application state) makes it a powerful and flexible option for building modern web applications.

Overall, Reacts role in the MERN stack is to provide a powerful and flexible library for building user interfaces, with a rich set of features and a large community of developers and resources to support it.

9.7 Self-Assessment Questions

Q1. What is full-stack web development and how does it differ from other types of web development? [6 marks, L2]

Q2. What is MERN? Explain the architecture of MERN with a suitable diagram. [8 marks, L2]

Q3. What are MERN components? Discuss in brief. [8 marks, L2]

Q4. Explain how ReactJS, Node.js, Express, and MongoDB work together in a MERN stack. [8 marks, L2]

Q5. What are some popular tools and libraries used in MERN development, and how do they facilitate the development process? [8 marks, L2]

Q6. What makes MERN a popular choice for web developers, and what advantages does it offer over other development stacks? [6 marks, L2]

Q7. How does JavaScript play a crucial role in MERN development, and what are some benefits of using JSON in a MERN stack? [4 marks, L2]

Q8. What are some factors that contribute to Node.js performance, and how does this impact MERN development? [4 marks, L2]

Q9. What is isomorphic JavaScript, and how does it relate to MERN development? [4 marks, L2]

Q10. What is React library and how does it fit into a MERN stack? [4 marks, L2]

9.8 Self-Assessment Activities

A1. Compare and contrast the MERN stack with other popular development stacks, such as MEAN or LAMP. What are the key differences between these stacks, and how do they impact the development process?

A2. Explore the various tools and libraries used in MERN development, such as Webpack, Redux, and Mongoose. How do these tools facilitate the development process?

A3. What are some best practices for developing applications using the MERN stack, and how can developers ensure optimal performance and scalability?

9.9 Multiple-Choice Questions

1. Full Stack Web Development is the _____. [1 mark, L1]
   a) Development of only the front end of a web application
   b) Development of only the back end of a web application
   c) Development of both the front end and backend of a web application
   d) Development of mobile applications only

2. MERN stand for _____. [1 mark, L1]
   a) MongoDB, Express, React, Node.js
   b) MySQL, Angular, React, Node.js
   c) MongoDB, Ember.js, React, Node.js
   d) MongoDB, Express, Redux, Node.js

3. ReactJS is _____. [1 mark, L1]
   a) A database management system
   b) A frontend library for building user interfaces
   c) A web application server
   d) A programming language

4. Node.js is _____. [1 mark, L1]
   a) A frontend library for building user interfaces
   b) A database management system
   c) A web application server
   d) A programming language

5. Express is _____. [1 mark, L1]
   a) A frontend library for building user interfaces

b) A database management system
c) A web application server
d) A programming language

6. MongoDB is _____. [1 mark, L1]
a) A frontend library for building user interfaces
b) A database management system
c) A web application server
d) A programming language

7. Which of the following is NOT a component of the MERN stack? [1 mark, L2]
a) ReactJS
b) Node.js
c) AngularJS
d) MongoDB

8. What is the purpose of tools and libraries in MERN development? [1 mark, L2]
a) To reduce development time and increase productivity
b) To add unnecessary complexity to the development process
c) To make the code harder to read and maintain
d) To make the development process more expensive

9. Why is MERN a popular choice for web development? [1 mark, L2]
a) Because it is optimized for single-page applications
b) Because it provides a high level of security and encryption
c) Because it allows for seamless integration with third-party APIs
d) Because it allows for rapid prototyping and development of full-stack applications

10. What is the purpose of using the React library in MERN development? [1 mark, L2]
a) Storing data in a database
b) Handling client-side routing
c) Building RESTful APIs
d) Building user interfaces

9.10 Key Answers to Multiple-Choice Questions

1. Full stack development is the development of both the front end and back end of a web application. [c]
2. MERN stands for MongoDB, Express, React, and Node.js. [a]
3. ReactJS is a frontend library for building user interfaces. [b]
4. Node.js is a Web Application Server. [c]
5. Express is a Web Application Server. [c]
6. MongoDB is a Database management system. [b]
7. AngularJS is NOT a component of the MERN stack.[c]
8. The purpose of tools and libraries in MERN development is to reduce development time and increase productivity. [a]
9. MERN is a popular choice for web development because it allows for rapid prototyping and development of full-stack applications. [d]
10. The purpose of using the React library in MERN development is for building user interfaces. [d]

9.11 Summary

Full stack development is the ability to develop web-based applications using both front-end skills and back-end skills. The front end refers to the client software and consists of the user interface that users see and interact with whereas the back end refers to the server software that takes care of the business logic that connects the application to the other services and the database. The back end of an application is responsible for data storage and retrieval and also for processing the requests sent to the server.

The front end uses technologies like HTML, CSS, and JavaScript. To create reusable components and manage the states of the applications front-end frameworks like React and Angular are most commonly used.

The back end uses server-side programming languages like Python, Node.js, Ruby on Rails, etc. To simplify the development process the most commonly used back-end frameworks are Express.js, Flask, and Ruby. For managing and storing data the databases commonly used are MongoDB, MySQL, PostgreSQL, Firebase, etc.

MERN is a widely used abbreviation that stands for MongoDB, Express.js, React.js, and Node.js.

- M in MERN stands for MongoDB. It is a type of database that is popular among developers. It's called a NoSQL (Non-Structured Query Language) database because it doesn't use a traditional query language.

- E in MERN stands for ExpressJS. It is a powerful tool that acts as a middleware and works with Node.js which is another technology in the MERN stack. It is used to help developers create web applications.
- R in MERN stands for ReactJS. It is a client-side JavaScript library that is used to create rich and interactive user interfaces for web applications.
- N in MERN stands for NodeJS. It is a runtime environment that is used to run JavaScript. Node.js is fast and efficient in code execution.

The MERN stack offers several benefits for web and mobile application development. They are – cost-effective, Search Engine Optimization (SEO) – friendly, Better performance, Improved security, Faster delivery, Faster modifications, Open-source, Easy switching, etc.

The MERN stack is typically used to implement a three-tier architecture in web development, also known as a client-server model.

The main components of MERN are - MongoDB, ExpressJS, ReactJS, and NodeJS.

The MERN stack is a popular choice for web developers because it provides a full-stack JavaScript framework for building modern, dynamic web applications.

To make development with MERN more efficient and effective, developers often utilize various tools and libraries specifically designed for each of these components. The purpose of tools and libraries in MERN development is to reduce development time and increase productivity.

9.12 Keywords

- MERN (MongoDB, ExpressJS, ReactJS, Node.js)
- MongoDB
- ExpressJS
- ReactJS
- Node.js
- Singel Page Applications (SPA)

9.13 Recommended resources for further reading

Text Book(s)

1. Vasan Subramanian, “Pro MERN Stack, Full Stack Web App Development with Mongo, Express, React and Node”, 2nd Edition, Apress.

2. Azat Mardan, 2017, “React Quickly_ Painless web apps with React, JSX, Redux, and GraphQL”, Manning Publications.

References

- Alex Banks and Eve Porcello, 2020, “Learning React Modern Patterns for Developing React Apps”, O'Reilly.
- Eddy Wilson, 2018, “MERN Quick start guide: Build Web applications with MongoDB, Express.js, React and Node”, Packt publishing.
- David Hows, 2014, Peter Membrey & Eelco Plugge, “MongoDB Basics”, Apress.
- Joe Morgan, ”How to code in React.js”, DigitalOcean.

--*--