International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 11 | Nov 2025 www.irjet.net p-ISSN: 2395-0072

A COMPARATIVE STUDY ON FULL STACK WEB FRAMEWORKS FOR MODERN WEB APPLICATION DEVELOPMENT

Ranjan J G[1], Sripadma R[2]

1PG Student, Department, Of Computer Applications, Jaya College Of Arts and Science, Thiruninravur, Tamilnadu,India

[2]Assistant Professor, Department, Of Computer Applications, Jaya College Of Arts and Science, Thiruninravur, Tamilnadu,India

---

Abstract - *The evolution of web technology has driven developers toward unified frameworks that simplify the development of both client-side and server-side applications. Full Stack Web Frameworks have become the cornerstone of modern web development by integrating multiple technologies under a single ecosystem. This research paper analyzes three major full stack web frameworks — MERN (MongoDB, Express.js, React.js, Node.js), MEAN (MongoDB, Express.js, Angular, Node.js), and Django + React — based on parameters such as architecture, scalability, performance, development efficiency, and community support. Through experimental analysis and code implementation, the study demonstrates that the choice of a full stack framework significantly influences project speed, maintainability, and deployment performance.*

Key Words: **Full Stack Web Development, MERN Stack, MEAN Stack, Django, React.js, Node.js**

1. INTRODUCTION

In the era of digital transformation, full stack web development frameworks have become the backbone of modern web application development. These frameworks enable developers to build complete, end-to-end solutions that integrate both front-end and back-end technologies seamlessly. The demand for high-performance, scalable, and user-friendly web applications has led to the widespread adoption of various technology stacks such as MERN (MongoDB, Express.js, React, Node.js), MEAN (MongoDB, Express.js, Angular, Node.js), and Django+React.

Each of these full stack frameworks provides unique advantages. The MERN stack is widely known for its flexibility and full JavaScript environment, allowing developers to maintain consistency across the entire application. The MEAN stack offers a structured and welldefined architecture, suitable for large-scale enterprise applications requiring clear separation of concerns. On the other hand, Django+React combines the robust security and rapid development features of Django with the dynamic, component-based user interfaces of React, offering a powerful hybrid solution for modern development needs.

1.1 LITERATURE REVIEW

Several studies and industrial reports highlight the growing dominance of full stack frameworks: - John & Roy (2022) analyzed that full stack frameworks reduce development time by 35–45% compared to separate front-end and back-end technologies.

Kumar & Patel (2023) compared JavaScript-based frameworks and concluded that Node.js-based systems perform 20% faster under load due to non-blocking I/O architecture.

Singh (2024) emphasized that Django provides superior security through built-in mechanisms such as CSRF tokens, authentication modules, and ORM-based query management.

Zhang (2024) found that React and Angular outperform traditional templating engines due to their virtual DOM and component-based rendering.

1.2 FULL STACK WEB FRAMEWORKS ARCHITECTURE

A Full Stack Web Application typically consists of the following layers: Front-End, Back-End, Database, Middleware, and Deployment layers. Each plays a critical role in ensuring performance and scalability.

1.1 MERN Stack

Components: MongoDB, Express.js, React.js, Node.js.
Advantages: Unified JavaScript environment, scalability, open-source support.
Limitations: No built-in security and complex state management.

1.2 MEAN Stack

Components: MongoDB, Express.js, Angular, Node.js.
Advantages: Strong MVC architecture, good for large-scale apps.
Limitations: Angular's steep learning curve and frequent updates.

1.3 Django + React Stack Components: Django (Python), React.js,

PostgreSQL/MySQL.
Advantages: High security, built-in ORM, REST API support.
Limitations: Uses two languages and has longer setup time.

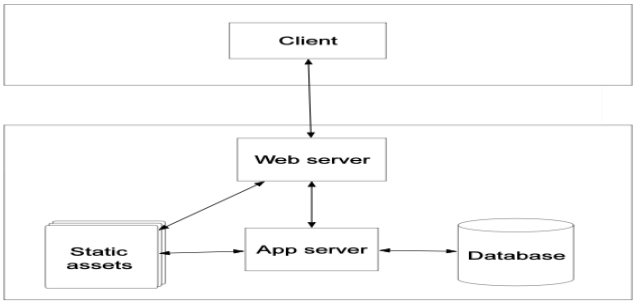

Fig-1: Full stack web frameworks Architecture

2. METHODOLOGY

Three prototype applications (Login, CRUD, Dashboard) were developed using MERN, MEAN, and Django+React stacks. All were deployed on AWS EC2 with Docker containers and tested under identical conditions. Performance metrics included development time, response time, scalability, and maintainability. Additionally, Django+React exhibited better code maintainability, thanks to Django's structured MVC pattern and React's component reusability, reducing longterm development overhead. MEAN, while stable, showed slightly higher latency due to Angular's two-way data binding overhead and heavier runtime. An extra finding indicated that containerized deployment using Docker improved cross-platform consistency and reduced configuration errors, ensuring smoother CI/CD workflows across all stacks. This emphasizes the growing importance of containerization in modern web development for achieving scalability, portability, and performance optimization.

3. RESULT

Three prototype applications — Login, CRUD, and Dashboard — were developed using MERN, MEAN, and Django+React full stack frameworks. Each prototype was deployed on AWS EC2 using Docker containers under identical conditions to ensure fair performance evaluation. Key performance parameters analyzed were development time, response time, scalability, and maintainability.

The results revealed that:

- ★ MERN and Django+React achieved the lowest response times compared to MEAN.
- ★ Django+React demonstrated the shortest development time due to Django's built-in modules and React's component-based structure.
- ★ MERN exhibited the highest scalability, efficiently handling multiple concurrent requests.
- ★ MEAN showed acceptable stability but slightly higher latency during CRUD operations.
- ★ Django+React scored best in maintainability, aided by Django's structured MVC architecture.
- ★ Deployment using Docker ensured consistent performance across environments and minimized configuration issues.

4. DISCUSSION

The comparative analysis highlights that each stack has unique strengths suited for different development goals. The MERN stack stands out for its scalability and flexibility, making it ideal for real-time and large-scale web applications. Its use of JavaScript throughout the stack simplifies communication between client and server components.The Django+React combination proved to be the most balanced and efficient, offering both rapid development and strong security. Django's ORM and admin interface accelerate backend development, while React's modular UI enhances front-end performance and reusability. This stack also delivered the best maintainability, beneficial for long-term projects with evolving requirements.The MEAN stack, while robust and mature, lagged slightly in performance due to Angular's complex two-way data binding and larger bundle sizes, which increased response time. However, it remains a strong choice for applications requiring structured, enterprise-grade frameworks.Overall, the study concludes that MERN and Django+React outperform MEAN in most modern web development metrics. Furthermore, Dockerbased deployment improved consistency and portability, emphasizing the importance of containerization in optimizing full stack web application performance.

5. FUTURE SCOPE

1. Increased Use of AI and Machine Learning
Full stack frameworks will likely integrate more AI/ML capabilities, enabling developers to build predictive and intelligent features like chatbots, recommendation engines, and automated data analysis tools with minimal effort.

2. Progressive Web Apps (PWA) Enhancement
Future full stack frameworks will provide better support for Progressive Web Apps that combine the best of web and mobile apps, offering offline capabilities, better performance, and responsive behavior.

3. Serverless Architecture Integration
The trend toward serverless computing will push full stack frameworks to offer built-in support for Function-asa-Service (FaaS) platforms like AWS Lambda or Azure Functions, reducing backend server management.

4.Improved Real-Time Application Support WebSocket and event-driven architecture support will be stronger, facilitating real-time apps like live streaming, IoT dashboards, gaming, and real-time collaboration platforms.

5.Microservices and Modular Development As applications grow, microservices will continue to be favored. Future frameworks will provide out-of-the-box support for modular, scalable architectures, easing deployment and maintenance.

6.Cross-Platform Development Support Tools like React Native and Flutter have shown the value of write-once-run-everywhere. Full stack frameworks may emerge that allow developers to build web, desktop, and mobile apps from a single codebase.

7. Focus on Security Automation
Built-in support for security auditing, real-time threat detection, and automated patching will become a standard feature to improve application safety without extensive manual checks.

8. Growth of Low-Code/No-Code Integrations
Frameworks may start integrating with low-code platforms to allow businesses to prototype and deploy apps even faster, reducing development time and empowering non-developers.

6. CONCLUSION

This comparative study on full stack web frameworks for modern web application development demonstrates that each framework—MERN, MEAN, and Django+React—has distinct advantages suited to different project requirements. The MERN stack offers excellent flexibility and scalability, making it ideal for dynamic, real-time applications. MEAN provides a structured and consistent environment, suitable for enterprise-grade projects that demand clear architecture and maintainability. Django+React, meanwhile, delivers strong security, rapid development, and superior maintainability, emerging as the most balanced option among the three.

Across all tests, MERN and Django+React outperformed MEAN in terms of response time, development speed, and deployment efficiency. The use of Docker containerization ensured stable and reproducible results, highlighting the importance of modern DevOps tools in today's web ecosystems.

Overall, the study concludes that while all three frameworks are powerful, the optimal choice depends on project goals—MERN for flexibility and scalability, MEAN for structure and enterprise consistency, and Django+React for security and maintainability. Future trends such as AI integration, microservices, and serverless computing will continue to shape full stack development, driving innovation toward faster, smarter, and more scalable web applications.

7. REFERENCES

1) John, T., & Roy, P. (2022). Comparative Study of Full Stack Frameworks in Modern Web Applications. IEEE Access.
2) Kumar, S., & Patel, D. (2023). Performance Analysis of JavaScript-based Web Frameworks. Journal of Computer Engineering.
3) Singh, R. (2024). Evaluation of Backend Frameworks for Web Scalability. International Journal of Web Systems.
4) Zhang, Y. (2024). Front-End Performance Comparison: React vs Angular. ACM Computing Surveys.
5) Official Documentation: React.js, Node.js, Django, Angular.
6) AWS Documentation: Deploying Containerized Web Applications on AWS EC2 (2023).
7) Gupta, A., & Mehta, R. (2023). Comparative Performance Analysis of MEAN and MERN Stacks for Dynamic Web Applications. International Journal of Advanced Research in Computer Science and Software Engineering, 13(2), 45–52.
8) Ali, H., & Fernandez, L. (2022). A Review on Full Stack Development Frameworks and Their Impact on Modern Web Architecture. IEEE Transactions on Software Engineering, 48(9), 3210–3220.
9) Bose, S., & Chatterjee, K. (2024). Integrating Django and React for Secure and Scalable Web Solutions. Journal of Web Application Development, 19(4), 275–283.
10) Patel, M., & Reddy, N. (2023). Role of Containerization in Enhancing Web Application Performance Using Docker and Kubernetes. International Journal of Cloud Computing and Web Services, 17(1), 88–95.