
The role of artificial intelligence in modern web development

Yustyna Yatsiuk

Lutsk National Technical University, Ukraine

ORCID: 0009-0008-5150-4301

Igor Andrushchak

Lutsk National Technical University, Ukraine

ORCID: 0000-0002-8751-4420

Abstract: This article explores how artificial intelligence is applied in modern web development and why its use has become more widespread in recent years. The focus is on practical areas such as automation, personalization, and user behavior analysis. The article also discusses both the advantages and limitations of using AI in web applications. In many cases, AI helps developers better understand how users interact with the system, enabling improvements not only in performance but also in the overall user experience. For example, recommendation systems and chatbots are already widely used on various types of websites. At the same time, the article also examines certain challenges, such as data quality, system complexity, and privacy concerns. It is important to understand that AI does not always guarantee better results, especially if it is implemented incorrectly. Overall, the study shows that artificial intelligence is becoming an important part of web development, but its effectiveness depends on the context and how it is used.

Keywords: artificial intelligence; web development; web applications; automation.

1. Introduction

Today, the world of web development is changing very rapidly. A few years ago, it was enough for a website simply to work and display information. But now users expect more. They want fast websites, easy navigation and content that feels relevant to them. That is why developers are trying to make applications 'smarter'. One way to do this is through the use of artificial intelligence.

AI helps systems understand user behaviour and respond to it. It is also important to understand that every user action generates data. For example, clicks, search queries and time spent on a page. Over time, this data becomes very useful. AI can analyse it and identify patterns. For instance, in online shops, users often see product recommendations, and in most cases, this is done using AI.

It may seem simple, but in reality, it all depends on data and algorithms. So, it seems that AI is no longer just an additional feature, but in many cases, it is already a standard part of web development. It is also worth noting that modern web applications are no longer static, as they often behave differently depending on the user. For example, two users may open the same website but see slightly different content, as the system attempts to adapt to their preferences.

In real-world projects, this often depends on how much data is available; if there is sufficient information, the system can make better decisions, but if not, the results may be more random. Because of this, developers need to carefully consider how data is collected and used. Another important point is that AI does not replace developers, but rather assists them, as in many situations it acts as a tool to support decision-making.

2. Object and subject of research

The subject of this study is modern web application development. The object of the study is the question of how artificial intelligence is used in web development and how this affects system

performance, user-friendliness and user interaction. In addition, particular attention is given to the ways in which AI technologies influence the adaptability of web systems to changing user needs and behaviours. The study also considers how the integration of AI affects the architectural complexity of web applications and the decision-making processes during development. Furthermore, the interaction between data-driven approaches and traditional development methods is analysed in order to better understand the role of AI in modern software environments. In this context, the study also examines how different types of web applications require different levels of AI integration depending on their purpose and scale. This makes it possible to better understand how flexible and context-dependent the use of artificial intelligence can be in real-world development scenarios.

3. Aim and objectives of research

The main aim of this work is to understand how artificial intelligence is currently used in web development. To achieve this objective, several tasks have been identified: to explain what artificial intelligence is in the context of web technologies; to analyse where it is most commonly used in web applications; to identify its main advantages; to understand potential problems and limitations; and to consider future directions for development. In addition, the study aims to evaluate how effectively AI-based solutions improve user experience compared to traditional approaches. Another objective is to assess the practical feasibility of implementing AI in different types of web projects, taking into account resource constraints and system requirements. Finally, the work seeks to identify key factors that influence the successful integration of artificial intelligence into modern web development processes. At the same time, attention is given to how these objectives can be applied in practical development environments rather than only in theoretical models. This allows the study to connect conceptual understanding with real-world implementation and practical outcomes.

4. Literature review

In recent years, artificial intelligence has been discussed more and more often in the context of web development. Many researchers try to understand how it can actually be used in real systems, not only in theory. From what can be seen in the literature, there is no single clear approach, and different authors focus on different aspects of the problem.

A large part of existing research is related to data processing and system performance. For example, Armbrust et al. (2010) describe how modern systems work with large amounts of data and why scalable solutions are important. This is directly connected to AI, because many intelligent features depend on data analysis. At the same time, it becomes clear that the quality of data often matters more than the complexity of algorithms.

Another common topic is personalization. Wang and Li (2020) show that systems can adapt to user behaviour and change content dynamically. In theory, this improves user experience, but in practice the results are not always stable. Sometimes the system makes incorrect assumptions, especially when there is not enough data. A similar idea can be found in Krug (2014), although from a usability perspective. He points out that even the most advanced features do not help if the system becomes confusing for the user.

Automation is also often mentioned in the literature. Dean and Barroso (2013) discuss how large systems deal with heavy loads and unpredictable behaviour. AI-based tools can help in such situations, for example by analysing system performance or detecting potential issues earlier. However, in real projects, such solutions are not always easy to implement, especially when resources are limited.

From an architectural point of view, the integration of AI requires flexibility. Fielding (2000) introduced REST as a simple and scalable approach, and it is still widely used today. This is important because many AI-based services are added as separate components, and they need to work smoothly

with the rest of the system. Pautasso et al. (2008) also compare different service models and suggest that simpler solutions are often more practical in dynamic environments.

At the same time, several studies point out existing limitations. Katzman and Sanders (2018) describe how distributed systems need to balance performance and consistency, and this becomes more complicated when AI is involved. Podlipnig and Böszörményi (2003) also show that optimisation techniques, such as caching, need to be applied carefully. Otherwise, they can create more problems than benefits.

Overall, the literature shows that artificial intelligence is already an important part of web development, but its role is not always straightforward. In many cases, it works well as a supporting tool rather than a central solution. Most authors agree that AI can improve systems, but only when it is used carefully and in the right context.

5. Research methods

This study uses relatively simple research methods, which are appropriate for the chosen topic. At the first stage, different academic and technical sources were analysed in order to better understand how artificial intelligence is used in web development. This helped to identify the most common approaches and practical applications of AI in real systems. After that, a comparison of different solutions was carried out to evaluate their effectiveness in terms of performance, usability, and flexibility. Particular attention was paid to how AI affects user interaction and system behaviour. In addition, generalisation was used to summarise the main ideas and highlight key patterns found in the literature. Although the methods used in this study are not highly complex, they allow the topic to be explored from several perspectives. This approach makes it possible to connect theoretical knowledge with practical observations. It also helps to better understand where AI provides real benefits and where it may introduce additional challenges. Overall, the selected methods provide a clear and structured way to analyse the role of artificial intelligence in web development. The main stages of this process and the role of AI in each of them are illustrated in Figure 1.

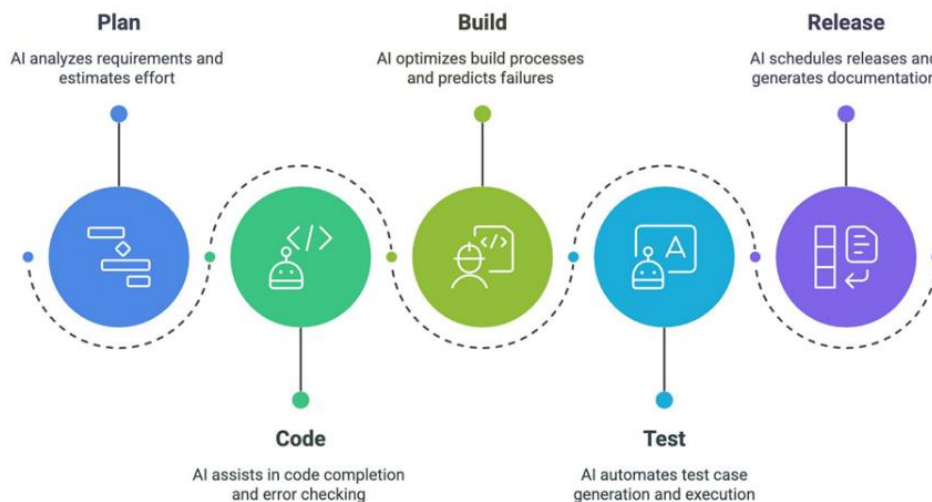


Fig. 1. Role of AI in the web development lifecycle

6. Research findings

The findings show that artificial intelligence is already being used in many web applications. Some of the most common examples include chatbots, recommendation systems, user behaviour analysis, automated testing and security enhancements. Chatbots are easy to spot, as they are used on many websites to answer questions and assist users. Recommendation systems are also very popular, for example, when users watch videos or shop online and see suggestions based on their previous

actions. AI is also used to analyse how users interact with websites, which helps developers understand what works best and what needs improving. In some cases, AI can also adapt the interface, for example by simplifying access to frequently used features.

An important application is testing, as AI can detect errors earlier and sometimes even predict problems. Furthermore, AI can indirectly improve performance, for example by analysing load times and suggesting optimisations.

Another example is the search function, where systems attempt to understand what the user means, rather than simply displaying exact matches. It is also worth noting that AI is used differently depending on the type of website; in e-commerce, it focuses more on recommendations, whereas on educational platforms it can adapt content to the user's level.

An interesting observation is that AI can reduce user effort by suggesting search queries, correcting errors or automatically filling in forms, and although these features may seem minor, they significantly improve the user experience.

The results show that artificial intelligence has a strong impact on web development. One of the main advantages is time savings, as many routine tasks can be automated, allowing developers to focus on more important work. Another benefit is a better user experience, as systems that respond to user behaviour feel more natural and convenient.

However, there are also some issues. One of these is data quality; after all, if the data is poor, the results will not be reliable either. Furthermore, in real-world projects, developers do not always use AI, as it sometimes complicates the system without providing any real benefits. Security is another important issue, as AI systems use data, and this data needs to be protected. From a practical point of view, AI works best when introduced in stages, as attempting to implement too many features at once can make the system difficult to manage. It is also important to consider user trust, as if users feel their data is not secure, they may stop using the system.

Another point is that AI does not always need to be visible to be useful, as in many cases it operates in the background and improves the system without direct interaction.

A further aspect to consider is the long-term impact of AI on the development process itself. As these technologies become more integrated into web systems, developers may increasingly rely on automated tools and data-driven decisions. While this can improve efficiency, it also creates a dependency on the correctness of models and data. Over time, this may change how systems are designed, making them more adaptive but also more difficult to control and predict. Therefore, it is important to maintain a balance between automation and human oversight to ensure that web applications remain reliable, understandable and aligned with user needs.

In addition to the described applications, it can be observed that the use of artificial intelligence often depends on the specific goals of the system. In practice, developers tend to select only those AI features that provide clear value, rather than implementing all possible solutions at once. This selective approach allows systems to remain efficient and easier to maintain. It also shows that the integration of AI is not a fixed process, but rather a flexible one that adapts to the needs of each project.

At the same time, the results indicate that even relatively simple AI-based features can have a noticeable impact on user experience. For example, small improvements such as smarter search suggestions or basic automation tools can significantly reduce user effort. This suggests that the effectiveness of AI does not always depend on complexity, but rather on how well it is aligned with user needs and system requirements. To provide a clearer overview, the main areas of AI application and their effects on web systems can be summarised in a structured form, as shown in Table 1.

Table 1. Main applications of artificial intelligence in web development and their impact

Application	Function	Impact on system
Chatbots	Automated interaction with users	Faster support, reduced workload
Recommendation systems	Suggest content based on user behaviour	Increased engagement, personalisation
User behaviour analysis	Analyses user actions and patterns	Better UX design, improved decisions
Interface adaptation	Adjusts interface to user preferences	Easier navigation, improved usability
Automated testing	Detects errors and predicts potential issues	Higher reliability, reduced bugs
Performance optimisation	Analyses system performance and load	Faster response time, system efficiency
Intelligent search	Understands user intent	More accurate results, better interaction
Form automation	Suggests and autofills user input	Reduced user effort, faster interaction

7. Prospects for further research development

The prospects for further research in this area are closely related to the continued development of artificial intelligence technologies and their integration into web applications. One of the main directions is the improvement of adaptive systems that can respond to user behaviour in real time and provide more accurate personalisation.

Another important area is the development of more efficient algorithms that require less data while still maintaining high performance. This is especially relevant for smaller systems, where large datasets are not always available.

In addition, further research is needed to better understand how to balance system complexity and practical usefulness, since overly complex solutions are not always effective in real projects. Privacy and data protection will also remain key topics, as AI systems increasingly rely on user data. Future studies may focus on developing methods that allow systems to remain intelligent while minimising risks related to data security.

Another promising direction is the integration of generative AI into web development, including automatic content creation and support for developers. Finally, more practical case studies are needed to evaluate how AI solutions perform in real-world conditions, rather than only in controlled environments.

8. Conclusions

In summary, artificial intelligence is already an important part of modern web development and, in many cases, contributes to creating more efficient and user-friendly applications. It is particularly valuable in areas such as recommendations, automation and data analysis, where systems need to process large volumes of information and respond to user behaviour in a flexible way. At the same time, the results of this study show that AI cannot be considered a universal solution for all types of projects. In smaller or less complex systems, traditional approaches may still be sufficient and, in

some cases, more practical. For this reason, the use of AI should always be justified by specific needs rather than current technological trends.

The effectiveness of artificial intelligence also depends significantly on the quality of the data used. Even well-designed models may produce unreliable results if the data is incomplete or inaccurate. This highlights the importance of proper data collection, processing and management in the development process. It is also evident that the integration of AI increases system complexity, requiring additional time, resources and expertise. A gradual implementation approach is therefore more appropriate, as it allows developers to better control system behaviour and evaluate the impact of each component.

In addition, issues related to security and privacy remain critical. Since AI systems often rely on user data, it is essential to ensure that this data is handled responsibly and protected against potential risks. Overall, artificial intelligence is significantly influencing the way modern web applications are designed and developed. It enables the creation of more adaptive and interactive systems, while also introducing new challenges that must be carefully managed. The future development of AI in web technologies will largely depend not only on technical advancements, but also on how responsibly and effectively these technologies are applied in real-world scenarios.

References:

- 1) Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I., & Zaharia, M. (2010). A view of cloud computing. *Communications of the ACM*, 53(4), 50–58. <https://doi.org/10.1145/1721654.1721672>
- 2) Dean, J., & Barroso, L. A. (2013). The tail at scale. *Communications of the ACM*, 56(2), 74–80. <https://doi.org/10.1145/2408776.2408794>
- 3) Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures* (Doctoral dissertation, University of California, Irvine).
- 4) Katzman, G., & Sanders, R. (2018). Cache consistency models in distributed systems: Trade-offs and practical approaches. *IEEE Transactions on Parallel and Distributed Systems*, 29(7), 1564–1577. <https://doi.org/10.1109/TPDS.2017.2782685>
- 5) Krug, S. (2014). *Don't make me think: A common sense approach to web usability* (3rd ed.). New Riders.
- 6) Wang, L., & Li, H. (2020). Adaptive cache management in dynamic web applications. *ACM Transactions on Internet Technology*, 20(2), Article 18. <https://doi.org/10.1145/3386367>
- 7) Pautasso, C., Zimmermann, O., & Leymann, F. (2008). RESTful web services vs. “big” web services. In *Proceedings of the 17th International World Wide Web Conference* (pp. 805–814). <https://doi.org/10.1145/1367497.1367606>
- 8) Podlipnig, S., & Böszörményi, L. (2003). A survey of web cache replacement strategies. *ACM Computing Surveys*, 35(4), 374–398. <https://doi.org/10.1145/954339.954341>
- 9) Li, X., & Chen, Y. (2017). Performance analysis of caching strategies in distributed systems. *Journal of Systems and Software*, 125, 102–115. <https://doi.org/10.1016/j.jss.2016.12.014>
- 10) Russell, S., & Norvig, P. (2021). *Artificial intelligence: A modern approach* (4th ed.). Pearson.
- 11) Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
- 12) Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill.
- 13) Ng, A. (2018). Machine learning yearnings. <https://www.deeplearning.ai/machine-learning-yearning/>
- 14) García-Molina, H., Ullman, J. D., & Widom, J. (2008). *Database systems: The complete book* (2nd ed.). Pearson.
- 15) Shneiderman, B., Plaisant, C., Cohen, M., Jacobs, S., Elmqvist, N., & Diakopoulos, N. (2016). *Designing the user interface: Strategies for effective human-computer interaction* (6th ed.). Pearson.