


Chapter 2


Leveraging Artificial Intelligence Techniques in Design Thinking Tools for Software Engineering

C. V. Suresh Babu

 <https://orcid.org/0000-0002-8474-2882>

Hindustan Institute of Technology and Science, India

Mala Raja Sekhar

 <https://orcid.org/0009-0007-6841-8330>

Hindustan Institute of Technology and Science, India

A. Sachin

Hindustan Institute of Technology and Science, India

Bala Brindha

Hindustan Institute of Technology and Science, India

ABSTRACT

This chapter explores the integration of AI techniques into design thinking tools for software engineering, aiming to enhance problem-solving, innovation, and user-centered development processes. The study focuses on software engineers, designers, and technology-driven organizations seeking to optimize software development workflows through AI-powered design thinking. The methodological approach includes a comprehensive review of existing AI applications in design thinking, an analysis of AI-enhanced prototyping and ideation tools, and case studies demonstrating their effectiveness in real-world software engineering scenarios. The findings reveal that

DOI: 10.4018/979-8-3693-9531-8.ch002

AI-driven design thinking tools streamline ideation, improve decision-making, and enhance user experience modeling, ultimately accelerating software development cycles. The chapter concludes that AI integration fosters a more adaptive, efficient, and innovative approach to software design, with significant implications for improving software engineering practices. .

1. INTRODUCTION

Software engineering has entered a transformative era where user-centric innovation and rapid adaptability are no longer luxuries but necessities. The evolving expectations of users, coupled with the increasing complexity of digital products, demand a rethink of conventional software development methodologies. This chapter begins by introducing the foundational ideas of design thinking and its growing influence in software engineering, especially in the context of rising AI integration. It also outlines the chapter's purpose and scope, preparing the reader for an exploration of how artificial intelligence is reshaping design thinking practices to meet modern software engineering challenges.

Design thinking is an empathetic approach that generates innovation and iterative testing to solve complex problems. This approach is applied in software engineering to create functional and aesthetically appealing solutions for users. Methodology Some of the stages involve empathizing with users, defining problems, and development of ideas, prototyping, and testing. Strength is in its iterative nature: design thinking reinforces itself in feedback loops, enhancing what should be high-quality products in software. Any landscape as rapidly shifting as that of software engineering—from a field whose users' expectations shift and evolve at once subtly and profoundly—design thinking offers a flexible structured framework. Traditional software development processes such as the Waterfall or Agile methodologies strongly orient towards technical performance and adhering to the deadline but miss the target when it comes to changing user requirements. Design thinking bridges this gap by placing the user at its center. Ingraining the concepts of design thinking into the processes of engineering software ensures that the products are both technologically robust and appealing to users (Buchanan, 1992).

The rise of Artificial Intelligence (AI) in software development signals a paradigm shift in how developers approach solving problems, efficiency, and innovation. AI-facilitated tools significantly impact the way developers write, test, debug, and optimize code. The integration of machine learning, NLP, and deep learning in recent times has transformed software engineering to be more dynamic. Through the automation of certain repetitive tasks, the developers can get assisted by insights into the products and even suggestions on how to enhance the code—that is leading

24 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/chapter/leveraging-artificial-intelligence-techniques-in-design-thinking-tools-for-software-engineering/382579

Related Content

Deep Learning-Based Tomato's Ripe and Unripe Classification System

Prasenjit Das, Jay Kant Pratap Singh Yadav and Laxman Singh (2022). *International Journal of Software Innovation* (pp. 1-20).

www.irma-international.org/article/deep-learning-based-tomatos-ripe-and-unripe-classification-system/292023

Enterprise Resource Planning Systems for Small and Medium-Sized Enterprises

Rogério Atem de Carvalho and Björn Johansson (2010). *Handbook of Research on Software Engineering and Productivity Technologies: Implications of Globalization* (pp. 373-381).

www.irma-international.org/chapter/enterprise-resource-planning-systems-small/37043

Fragmentation of Mobile Applications

Damith C. Rajapakse (2012). *Handbook of Research on Mobile Software Engineering: Design, Implementation, and Emergent Applications* (pp. 317-335).

www.irma-international.org/chapter/fragmentation-mobile-applications/66475

Innovation in Curriculum System for Communication Majors Based on New Media Technology and Big Data Analytics

Xiaofei Chen (2025). *International Journal of Information System Modeling and Design* (pp. 1-26).

www.irma-international.org/article/innovation-in-curriculum-system-for-communication-majors-based-on-new-media-technology-and-big-data-analytics/391367

Deep Analysis of Unbalanced Offline Data in Bridge Monitoring Based on Improved Upper Confidence Bound Algorithm

Lei Fu and Hongju Han (2025). *International Journal of Information System Modeling and Design* (pp. 1-26).

www.irma-international.org/article/deep-analysis-of-unbalanced-offline-data-in-bridge-monitoring-based-on-improved-upper-confidence-bound-algorithm/374215