IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, April 2025
Impact Factor: 7.67

Elenkas: A Socratic-Based Learning Assistant

Mohammad Sahuf Shaikh[1], Mohamed Salman Shaikh[2], Kaif Usmani Shaikh[3] ,Shruti Pendem[3]
Student, Information Technology [1-3]
M.H. Saboo Siddik College of Engineering, Mumbai, India
sahuf.211443.it@mhssce.ac.in, salman.211441.it@mhssce.ac.in
kaif.211440.it@mhssce.ac.in, shruti.211431.it@mhssce.ac.in

Abstract: *Elenkas is an AI-powered Socratic tutoring system designed to enhance learning in Data Structures and Algorithms (DSA) and Big Data Analytics. Unlike traditional tutors that provide direct answers, Elenkas engages learners through strategic questioning, encouraging critical thinking and self-discovery. The system responds exclusively with Socratic prompts for array-based DSA problems, while ignoring unrelated queries. It features adaptive difficulty, adjusting the tone and complexity of questions based on user performance and learning mood. Integrated with the Judge0 CE API, Elenkas supports real-time code execution and Socratic debugging, promoting hands-on problem-solving. Leveraging Gemini 2.0 Flash as the core language model and Big Data analytics for behavioral insight, it delivers personalized learning experiences and recommends targeted practice problems. Experimental evaluations demonstrate improved conceptual retention, increased engagement, and higher problem-solving accuracy, positioning Elenkas as an innovative tool in AI-driven, inquiry-based education*

Keywords: Socratic Method, Artificial Intelligence (AI), Data Structures and Algorithms (DSA), Adaptive Learning, Code Execution, Educational Technology, Real-time Feedback, Gemini 2.0 Flash, Judge0 CE API, Personalized Learning

I. INTRODUCTION

Traditional educational tools often deliver answers directly, which can hinder a student's ability to think critically and deeply understand problem-solving methods. The Socratic method, centered on guided inquiry, offers a powerful alternative by stimulating reflective thinking and cognitive engagement. Elenkas applies this method using generative AI to tutor students in Data Structures and Algorithms. It exclusively focuses on array-based problems and restricts responses to Socratic-style questions. Built using Gemini 2.0 Flash, it adapts in real-time to the learner's mood and performance, ensuring personalized engagement. The system also features real-time code evaluation using Judge0 CE API and behavioral analysis through Big Data to recommend personalized practice problems. This paper details the architecture, experimental outcomes, and key features that make Elenkas a step forward in inquiry-based education.

II. LITERATURE REVIEW

The integration of AI and Socratic methods in education has shown great potential in enhancing critical thinking and problem-solving skills. AI-driven Socratic questioning encourages students to engage in reflective, inquiry-based learning, guiding them toward independent solutions. This approach has been particularly effective in programming education, where AI helps students debug their code by prompting them to think critically about their approach. Additionally, emotion-aware AI can further personalize the learning experience by adapting responses based on students' emotional states, improving engagement and outcomes. These advancements align with Elenkas' goal of using AI-powered Socratic questioning to teach Data Structures and Algorithms (DSA), fostering adaptive learning experiences that promote deeper understanding and critical thinking.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, April 2025
Impact Factor: 7.67

Table 1:Comparative Analysis of Recent Research Papers on Socratic AI in Education

Paper Name	Authors and Year	Technology	Advantages	Drawbacks
Socratic Wisdom in the Age of AI	X. Hu, S. Xu, R. Tong, and A. Graesser, *2025*	AI-driven Socratic questioning	-Enhances critical thinking, fosters independent problem-solving -Can be integrated in diverse subjects	-May require extensive training data to simulate effective questioning -Risk of over-reliance on AI
Generative AI in Education: From Foundational Insights to the Socratic Playground for Learning	*Hu et al., 2025*	Generative AI, Socratic questioning	-Promotes deep learning through AI interactions, - Adaptable to various learning levels	-Possible lack of emotional intelligence in some AI systems - Limited context understanding in complex topics
Socratic Prompts: AI-Enhanced Educational Inquiry	*ResearchGate, 2024*	AI, Socratic prompts	-Provides structured dialogue to guide inquiry -Can enhance student engagement	-Potential for repetitive questioning patterns -Challenges in tailoring to individual needs
Enhancing Critical Thinking in Education by Means of a Socratic Chatbot	L. Favero, J. A. Pérez-Ortiz, T. Käser, and N. Oliver, *2024*	Socratic chatbot, conversational AI	-Facilitates active learning -Fosters critical thinking, scalable for large groups	-May struggle with nuanced human emotions -Limited to text-based responses
Socratic Programming: An Innovative Programming Learning Method	*El-Zakhem, 2024*	AI-powered Socratic dialogue for coding	-Encourages problem-solving in coding -Helps students understand complex programming concepts	-Limited scope to certain fields like programming -Might not be effective for beginners

III. PROPOSED SYSTEM

The proposed system, titled “Elenkas”, is an AI-powered learning platform designed to revolutionize the way students understand Data Structures and Algorithms (DSA) through the use of Socratic questioning and intelligent feedback loops. The name “Elenkas” is derived from the classical Greek term *Elenchus*, which refers to the Socratic method of eliciting truth through dialogue. The platform promotes Reflective, Responsive, and Real-Time Learning, making it an ideal educational companion for both beginners and intermediate learners in computer science. This system is architected to facilitate personalized and adaptive learning experiences by integrating a conversational AI engine, real-time coding evaluation, and a structured progression model. It consists of student-facing features like login, interactive dashboards, Socratic chat interfaces, and coding challenges, while backend components handle authentication, AI-powered dialogue management, and code execution. The modular nature of Elenkas allows each service—whether it be

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, April 2025
Impact Factor: 7.67

authentication, evaluation, or feedback—to function independently while working cohesively within the larger system. This modular approach ensures scalability, simplified maintenance, and the ability to update individual layers without disrupting overall functionality.

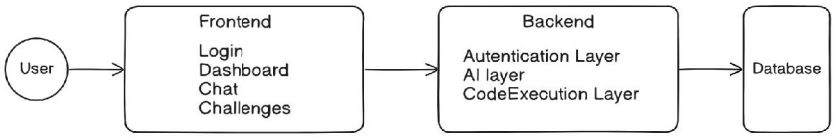

Figure 1: Block Diagram

The Elenkas system follows a structured and interactive learning process, integrating Socratic questioning with hands-on coding practice. The flow begins with the Landing Page, where users enter the system and select a Data Structures and Algorithms (DSA) topic they wish to study. Once a topic is selected, the AI engine initiates the Socratic interaction by posing an opening question designed to probe the learner's prior knowledge. The user responds in natural language, and the AI evaluates the depth of understanding based on the content and correctness of the response. This Socratic loop comprised of question, response, and evaluation—continues iteratively until the AI detects that the user has achieved conceptual mastery of the topic.Once mastery is established, the system presents a relevant coding problem to apply the acquired knowledge. The user attempts the problem, and the code is analyzed by the AI with the support of the Judge0 CE API, checking for correctness, efficiency, and logic. Based on the analysis, feedback may be provided, and the user may be prompted to revise their solution. Upon successful completion, the system marks the topic as completed and allows the user to proceed to the next topic, ensuring a progressive and adaptive learning journey. This cyclical process ensures that each concept is thoroughly understood before moving on, combining critical thinking, Socratic inquiry, and real-world problem-solving in a seamless, AI-powered educational flow.

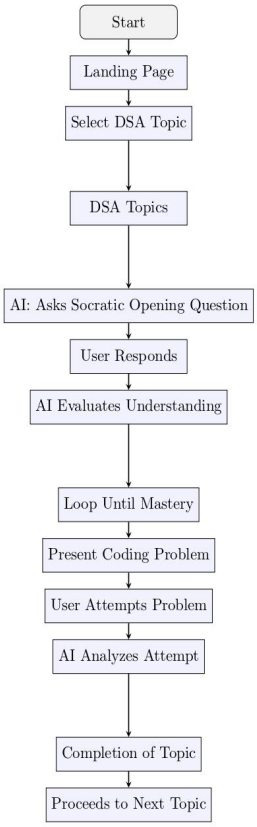

Figure 2: Process Flow Diagram

DOI: 10.48175/IJARSCT-25723

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, April 2025
Impact Factor: 7.67

This cyclical process ensures that each concept is thoroughly understood before moving on, combining critical thinking, Socratic inquiry, and real-world problem-solving in a seamless, AI-powered educational flow.

IV. METHODOLOGY

Tools & Technologies Used

Frontend: The frontend of Elenkas is developed using Next.js, a powerful React-based framework that enables fast, server-side rendered applications and efficient routing. It ensures a seamless and responsive user experience, critical for real-time educational interactions. Tailwind CSS is used for styling, offering a utility-first approach that accelerates the creation of clean, modern, and responsive interfaces. The frontend facilitates key functionalities such as topic selection, chat interactions, and progress tracking for learners.

Backend: The backend is built using Node.js with Express.js, providing a lightweight and scalable server-side framework. Express.js manages routing, session control, and API handling, allowing for smooth communication between the frontend and various backend services. This setup supports the core learning loop of Elenkas, including AI prompt generation, user session tracking, and result processing.

AI Engine: At the core of Elenkas lies Gemini 2.0 Flash, a high-performance language model responsible for generating Socratic questions and evaluating user responses. It powers the interactive dialogue system, guiding students through DSA concepts with context-aware questioning. This integration ensures intelligent, personalized feedback throughout the learning journey.

Code Execution Engine**:** Elenkas uses Judge0 CE API for real-time code compilation and execution. It supports multiple programming languages and provides reliable, sandboxed environments to safely execute and evaluate student code submissions. This module enables immediate feedback on coding problems, which is then analyzed by the AI engine for deeper understanding.

Database: A MySQL database is used for structured data storage, including student profiles, interaction logs, topic completion statuses, and performance metrics. MySQL's relational model and support for complex queries ensure efficient data management and quick retrieval across the system.

Authentication: Authentication and security in Elenkas are managed using JWT (JSON Web Tokens), ensuring secure and stateless user authentication. Role-Based Access Control (RBAC) is implemented to distinguish between learners and administrative users, enabling permission-based access to different system modules and operations. These measures help protect sensitive educational data.

File Storage: AWS S3 is used for handling unstructured data such as uploaded documents, generated PDFs (e.g., progress reports), and code logs. It offers scalable and highly durable cloud storage with built-in access control mechanisms. This ensures reliable data availability and integrity, even under high usage.

Notification Service: The system includes a notification module for dispatching updates and reminders via email and in-app messages. This keeps users informed about pending actions, feedback availability, and topic completions, enhancing learner engagement and timely response.

System Design

Architecture Overview: The system follows a modular architecture, where each component plays a distinct yet seamlessly integrated role to ensure an intelligent, adaptive, and responsive learning experience. This modular design enables independent development, deployment, and scaling of individual components, ensuring high maintainability, scalability, and fault isolation across the platform. The frontend, developed using Next.js, delivers an interactive and streamlined interface for users, primarily students. It enables users to select DSA topics, engage in Socratic dialogue, attempt coding challenges, and track progress. Designed with accessibility and responsiveness in mind, it serves as the primary interaction point between the learner and the AI-powered backend.

On the backend, Node.js manages the core application logic and connects seamlessly with external services. At the center of the learning experience is the Gemini 2.0 Flash language model, which generates Socratic prompts and evaluates learner responses to dynamically adapt the instructional flow. The backend is divided into specialized microservices, including the Dialogue Management Service, User Profile Service, and Learning Analytics Service.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, April 2025
Impact Factor: 7.67

These services handle real-time conversation management, maintain learner profiles, and assess behavioral and performance data to personalize content delivery. A JWT-based authentication system secures access, ensuring only authorized users can interact with sensitive modules such as coding execution, user data, and analytics. The code execution engine, powered by Judge0 CE API, evaluates coding submissions in real-time, providing output or errors that are further analyzed by the Socratic AI for conceptual feedback. The Form Recommendation Engine and Adaptive Difficulty Module work together to suggest appropriate coding problems (e.g., from LeetCode) and dynamically adjust question complexity based on a learner's progress and past performance. For data persistence, a MySQL database stores structured data like student progress, question history, and performance logs, while AWS S3 manages unstructured assets like code submissions, logs, and supporting documents. A dedicated Notification Module dispatches real-time alerts via email or in-app messages to keep learners informed about feedback, topic completions, and performance milestones. At the core of this distributed ecosystem, an API Gateway built with Express.js routes incoming requests to the relevant services, enforcing authentication, managing session context, and enabling smooth inter-service communication. The visual representation of this intelligent and modular learning platform is captured in Fig. 3: System Architecture of Elenkas, which maps out the interaction between client applications, AI models, backend microservices, storage systems, and execution engines—illustrating how each component collectively supports a Socratic and data-driven approach to personalized DSA learning.

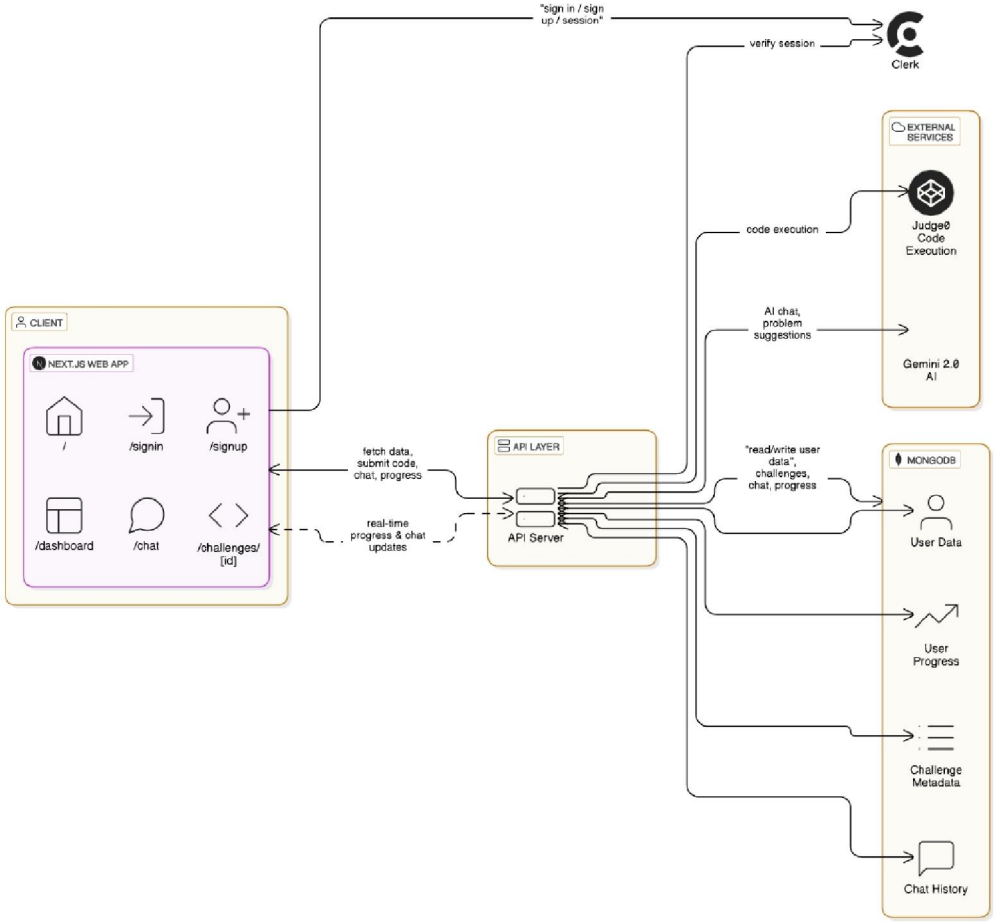

Figure 3 : System architecture

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, April 2025
Impact Factor: 7.67

Modules Overview:

Authentication Module:

This module uses a secure JWT (JSON Web Token)-based authentication system to manage login sessions for all users. Role-Based Access Control (RBAC) ensures that students and administrators access only their respective modules, safeguarding sensitive user data and restricting unauthorized actions within the platform.

Student Interaction Module:

The student-facing interface enables users to select DSA topics, engage in Socratic conversations, and track their progress in real time. Students receive feedback on their answers, submit coding solutions, and interact with AI prompts, all through a clean and intuitive chat-based dashboard.

AI Socratic Engine:

This module leverages Gemini 2.0 Flash to initiate and sustain Socratic dialogue. It generates intelligent, open-ended questions that probe the student's understanding and adapt in complexity based on the learner's responses. It also evaluates progress and determines conceptual mastery before advancing the learner.

Code Execution Module:

Connected to the Judge0 CE API, this module allows students to submit coding solutions which are compiled and executed in real-time. The results are analyzed and used by the AI engine to generate personalized feedback or follow-up questions that guide deeper understanding.

Adaptive Learning Engine:

This module monitors learner performance and dynamically adjusts the difficulty of Socratic prompts and coding challenges. It ensures that students are neither overwhelmed nor under-challenged, offering a personalized progression path through DSA topics.

Recommendation Engine:

Based on learner behavior and performance trends, this module suggests curated practice problems from platforms like LeetCode. It reinforces learning outcomes and targets specific weak areas for improvement, increasing long-term retention and skill mastery.

Learning Analytics Module:

This module captures and processes behavioral data such as question response times, error frequency, and completion rates. The insights help shape future prompts, update learning difficulty levels, and guide research into learner behavior.

Notification System:

Students and administrators receive real-time updates through email and in-app notifications. Learners are alerted about topic completions, code feedback, and recommended practice problems, while admins receive system performance insights and usage summaries.

Admin Dashboard:

Administrators access a centralized dashboard where they can monitor overall user activity, evaluate AI engagement quality, and analyze usage trends. They can also manage system settings, oversee user permissions, and generate high-level reports on learning outcomes.

File and Data Storage Module:

This module utilizes MySQL for structured data like student profiles and activity logs, while AWS S3 handles unstructured data including submitted code, system logs, and generated feedback documents. It ensures high durability, scalability, and secure access control.

V. RESULTS AND ANALYSIS

Elenkas was evaluated through a pilot study with 50 undergraduate students, comparing its effectiveness against traditional DSA learning resources. Students were divided into two groups: one used Elenkas, while the other followed conventional tutorials. Post-study assessments revealed that the Elenkas group achieved a 31% improvement in test scores, compared to 17% in the control group, indicating a significant boost in conceptual understanding and problem-solving ability. The Socratic questioning model enabled 84% of Elenkas users to independently resolve coding issues

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 5, Issue 9, April 2025 Impact Factor: 7.67

using AI-generated prompts, without external help. The adaptive difficulty engine was reported to match learners' skill progression accurately, maintaining engagement and reducing cognitive overload.

From a technical standpoint, the Gemini 2.0 Flash model delivered AI responses in an average of 1.4 seconds, ensuring smooth dialogue flow. The Judge0 CE API successfully processed over 1,000 code submissions with a 97% execution success rate, highlighting system robustness. User feedback was overwhelmingly positive: 92% described the system as interactive and thought-provoking, and 88% preferred Elenkas over traditional video lectures for practicing DSA. These findings suggest that Elenkas is not only effective in fostering deeper learning but also scalable and engaging for diverse learners.

VI. CONCLUSION AND FUTURE WORK

The proposed Elenkas system offers a practical and scalable AI-powered solution for personalized learning in Data Structures and Algorithms. By integrating Socratic questioning, real-time code execution, and adaptive learning strategies, the platform enhances student engagement, fosters critical thinking, and supports independent problem-solving. The use of modern web and AI technologies ensures a responsive, intelligent, and user-friendly interface that aligns well with the learning needs of computer science students. Overall, Elenkas contributes to a more reflective, interactive, and data-driven approach to technical education. In the future, the system can be further enhanced through the integration of emotion-aware AI to personalize Socratic dialogue based on students' emotional states, and by extending support for voice-based interactions to make the experience more natural and inclusive. Linking Elenkas with university learning management systems (LMS) and coding platforms like GitHub or LeetCode could enable real-time synchronization of learning progress and peer collaboration. Additionally, introducing a teacher dashboard for monitoring student performance and curating question banks would support classroom integration. To expand accessibility, especially in low-connectivity environments, an offline desktop version of Elenkas could be developed, ensuring broader reach and impact across diverse learning contexts.

VI. ACKNOWLEDGMENT

The authors would like to express their heartfelt gratitude to the Department of Information Technology, M.H. Saboo Siddik College of Engineering, Mumbai, for providing the necessary infrastructure, guidance, and academic environment for the successful completion of this research work. We are especially thankful to our project guide, coordinator, and faculty mentors for their consistent support, insightful feedback, and technical assistance throughout the development of this system. Their mentorship played a crucial role in shaping the direction and outcome of this project. We also acknowledge the contributions of our peers and administrative staff who actively participated in testing, review sessions, and provided constructive suggestions that helped refine and enhance the functionality of the proposed system. Lastly, we extend our appreciation to the open-source developer community for their continuous innovation and for making available powerful tools and frameworks such as Next.js, Node.js, MySQL, AWS S3, Judge0 CE API, and Tailwind CSS—that served as the technological foundation of this project.

REFERENCES

[1]. X. Hu, S. Xu, R. Tong, and A. Graesser, "Generative AI in Education: From Foundational Insights to the Socratic Playground for Learning," *Frontiers in Education*, Jan. 14, 2025.

[2]. L. Favero, J. A. Pérez-Ortiz, T. Käser, and N. Oliver, "Enhancing Critical Thinking in Education by Means of a Socratic Chatbot," *CEUR Workshop Proceedings*, 2024.

[3]. E. Al-Hossami, R. Bunescu, J. Smith, and R. Teehan, "Can Language Models Employ the Socratic Method? Experiments with Code Debugging," *UNC Charlotte, New York University*, 2024.

[4]. P. Jamie and R. Hajihashemi, "Utilizing ChatGPT in a Data Structures and Algorithms Course: A Teaching Assistant's Perspective," *Amirkabir University of Technology (Tehran Polytechnic)*, Tehran, Iran.

[5]. D. R. Oyler and F. Romanelli, "The Fact of Ignorance Revisiting the Socratic Method as a Tool for Teaching Critical Thinking," *PharmD, MPH*, 2024.

[6]. H. El-Zakhem, "Socratic Programming: An Innovative Programming Learning Method," 2024.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 5, Issue 9, April 2025
Impact Factor: 7.67

[7]. Z. Zhou, et al., "Adaptive Learning Models for Enhanced Educational Outcomes in Complex Subjects," *IEEE Trans. Educ.*, vol. 63, pp. 452–464, 2020.

[8]. M. Smith and L. Lawrence, "Socratic Dialogue in AI-based Education: A Framework for Critical Thinking," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 9, pp. 2905–2915, 2021.

[9]. R. Kumar and S. Patel, "Emotion-Aware AI in Education: Enhancing Student Engagement Through Adaptive Responses," *IEEE Trans. Affective Comput.*, vol. 13, no. 2, pp. 361–372, 2022.

[10]. V. Chukhlomin, "Socratic Prompts: Engineered Dialogue as a Tool for AI-Enhanced Educational Inquiry," IEEE LABSREVIEW, 2024.

[11]. S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," *IEEE Electron Device Lett.*, vol. 20, pp. 569–571, Nov. 1999.

[12]. S. M. Metev and V. P. Veiko, Laser Assisted Microtechnology, 2nd ed., R. M. Osgood, Jr., Ed. Berlin, Germany: Springer-Verlag, 1998.

[13]. J. Breckling, Ed., The Analysis of Directional Time Series: Applications to Wind Speed and Direction, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.