

DCE TECH CHRONICLE

— IDEAS | INNOVATION | TECHNOLOGY —

CSE EDITION

Volume VI, Issue I
(July - December 2025)



CONTENTS

CSE EDITION | VOLUME VI, ISSUE I
(JULY - DECEMBER 2025)

01



HOD Message

02



Editor's Message

03



Vision & Mission

04



PEOs, POs, PSOs

05



Student's Articles

06



**Department
Achievements**



HOD's MESSAGE

Dr. Ashima Mehta
(HOD CSE)



“Technology does not simply change the world; it empowers us to reimagine what is possible.”

It is a matter of great pride to introduce this technical magazine, a reflection of the ideas, imagination, and innovative spirit of our Computer Science & Engineering community. In an era where Artificial Intelligence, Data Science, Cybersecurity, Cloud Computing, and emerging technologies are redefining the way we live and work, the ability to learn, question, create, and innovate has become more important than ever.

This magazine is more than a collection of technical articles—it is a glimpse into the curiosity and potential of our students. Every idea shared, project explored, and insight presented represents a step towards becoming not just technology users, but technology creators and problem-solvers.

As a department, our endeavour is to provide an environment where students can experiment with ideas, embrace challenges, learn from failures, and transform their knowledge into meaningful solutions. I encourage our students to remain curious, keep exploring beyond the classroom, and have the courage to turn their ideas into reality.

My appreciation goes to the faculty members, students, and editorial team whose dedication has brought this edition to life. I hope these pages spark new thoughts, inspire innovation, and encourage every reader to ask one important question: “What can I create that has never been created before?”

Wishing the CSE community a future filled with curiosity, creativity, and limitless possibilities.

Dr. Ashima Mehta
Head, Department of Computer Science & Engineering

Editor's Message



Editor in-Chief

It gives me immense pleasure to present this edition of our Technical Magazine. This platform showcases the creativity, curiosity and technical spirit of our students and faculty. I hope it inspires every reader to learn, explore, innovate and create.

Prof. Vimmi Malhotra
(Faculty; CSE Department)

STUDENT EDITORS



Swayam Prabha

CO Editor

Roll No.: 26184; 3rd year



Veer Bhadra Singh

Editor (Design)

Roll No.: 26195; 3rd year



Kanishka

Editor (Text)

Roll No.: 26081; 3rd year

Small Idea, Big Impact



Vision

Steering the future of computer science through innovative advancements, fostering ethical values and principles through technical education.



Mission

- Directing future innovations in computer science through revolutionary progress.
- Instilling a foundation of ethical values and principles in every technologist.
- Offering a comprehensive technical education to equip individuals for a meaningful and influential future.

Program Educational Objectives (PEO)

- Apply the technical competence in Computer Science and Engineering for solving problems in the real world.
- Carry out research and develop solutions on problems of social applications.
- Work in a corporate environment, demonstrating team skills, work morals, flexibility and lifelong learning.

Program Specific Outcome (PSO)

- Exhibit design and programming skills to develop and mechanize business solutions using revolutionary technologies.
- Learn strong theoretical foundation leading to brilliance and enthusiasm towards research, to provide well-designed solutions to complicated problems.
- Work effectively with diverse Engineering fields as a team to design, build and develop system applications.

Programme Outcome (Pos)

- **PO1:** Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.
- **PO2:** Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)
- **PO3:** Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)
- **PO4:** Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).
- **PO5:** Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)
- **PO6:** The Engineer and The World: Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).
- **PO7:** Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)
- **PO8:** Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.
- **PO9:** Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences
- **PO10:** Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.
- **PO11:** Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change.(WK8)

Student's Articles

Beyond Traditional Coding: The Rise of AI-Assisted Programming



Akriti Kumari
Roll. No. : 26020(3rd Year)

Programming has traditionally required developers to translate a problem into algorithms, write code, test it, identify errors, and refine the solution. With the rapid development of Artificial Intelligence (AI), this process is changing. AI-assisted programming allows developers to work with intelligent tools that can generate code, suggest improvements, explain errors, create test cases, and assist with documentation.

Rather than replacing programmers, AI is becoming a **development partner** that can handle repetitive tasks while allowing programmers to concentrate more on problem-solving, system design, and innovation.

From Code Writing to Problem Solving

In traditional programming, developers manually write most of the code required for an application. AI-assisted programming introduces a different approach. Developers can describe a requirement in natural language, and an AI tool can suggest a possible implementation.

For example, instead of manually creating a function to search and filter a large dataset, a programmer can describe the required inputs, outputs, and conditions. The AI can generate an initial solution that the programmer can review, modify, and test.

Student's Articles

This changes the role of a programmer from simply writing code to **designing solutions and validating machine-generated code**. Programming fundamentals remain essential because developers need to understand whether the suggested solution is logically correct, efficient, secure, and suitable for the application.

Major Applications of AI-Assisted Programming

1. Code Generation and Completion

AI tools can generate complete functions or suggest the next lines of code while a developer is programming. This can reduce repetitive coding and help developers quickly create initial implementations.

2. Debugging

Finding programming errors can consume significant development time. AI systems can analyse error messages and code and suggest possible causes and solutions. They can also explain complex code in simpler terms, making debugging more accessible to students and developers.

3. Code Refactoring

As applications become larger, code can become repetitive or difficult to maintain. AI can identify opportunities to simplify code, remove unnecessary repetition, and reorganise functions while attempting to preserve the original behaviour.

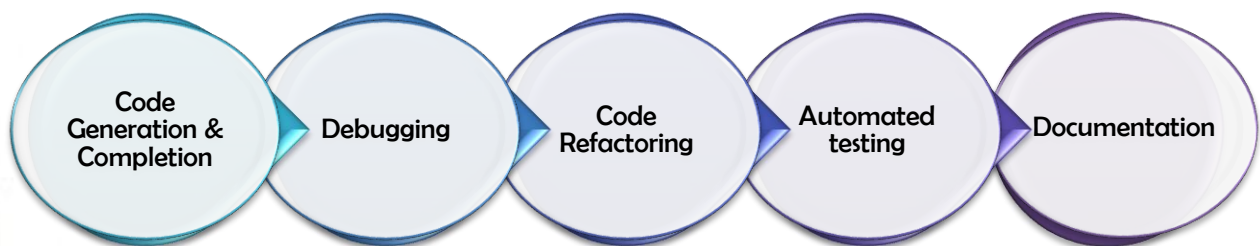
Student's Articles

4. Automated Testing

AI can generate potential test cases based on program logic. It can suggest normal, boundary, and unusual inputs that developers may use to identify defects. However, developers must still verify whether the generated tests adequately represent real-world requirements.

5. Documentation

Understanding existing software can be difficult, particularly when working with large or older codebases. AI can assist in generating comments, explaining functions, summarising modules, and preparing basic technical documentation.



Natural Language as a Programming Interface

One of the most significant developments is the use of **natural language for interacting with programming tools**. Developers can describe what they want instead of initially expressing every detail through programming syntax.

For example, a developer might request:

“Create a Python function that accepts a list of numbers, removes duplicate values, sorts the remaining values, and returns the result.”

Student's Articles

An AI system can produce a possible implementation almost immediately.

However, the quality of the result depends significantly on the clarity of the instructions. Developers therefore need to learn how to provide precise requirements, constraints, examples, and expected outputs. This has made effective communication with AI an emerging skill in software development.

Why Human Expertise Is Still Essential

AI-generated code can appear correct while containing logical errors, inefficient algorithms, outdated methods, or security weaknesses. An AI system may also misunderstand the specific requirements of an organisation or application. For example, code generated for a student-management application might technically function but fail to consider privacy, authentication, invalid inputs, or institutional rules. Therefore, programmers must continue to perform **code review, testing, security analysis, performance evaluation, and requirement validation**. The responsibility for the final software remains with the human development team.

AI should therefore be viewed as an assistant rather than an authority.

Student's Articles



Challenges of AI-Assisted Programming

Despite its advantages, AI-assisted programming presents several challenges.

Accuracy: Generated code may contain errors or incorrect assumptions.

Security: AI-generated solutions can unintentionally introduce vulnerabilities if they are not carefully reviewed.

Overdependence: Students who use AI for every programming problem may fail to develop strong algorithmic and problem-solving skills.

Maintainability: A generated solution may work for a particular example but may not be efficient or scalable for a larger application.

Student's Articles

Data Privacy: Developers must be careful when providing proprietary source code, credentials, personal information, or confidential project details to external AI tools. These challenges highlight the importance of responsible and informed use of AI in software development.

The Programmer of the Future

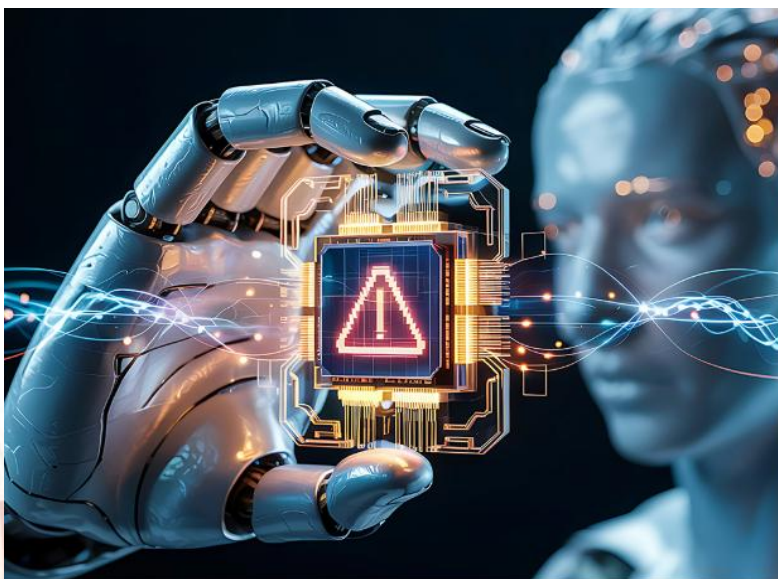
The emergence of AI-assisted programming is not making programming knowledge obsolete instead, it is changing the skills required from programmers.

Future developers will need a combination of:

Programming Fundamentals + Algorithmic Thinking + AI Literacy + Testing + Critical Evaluation

Students should therefore use AI to enhance learning rather than replace it. A productive approach is to first understand a problem, attempt a solution, use AI to explore alternative approaches, compare the results, and finally test and improve the chosen implementation.

This approach develops both programming ability and AI literacy.



Student's Articles

In Conclusion, AI-assisted programming represents a major evolution in software development. AI can reduce repetitive coding, accelerate debugging, support testing, and help developers understand complex software. At the same time, generated code must be carefully reviewed because AI does not automatically guarantee correctness, security, or efficiency.

The future of programming is therefore not simply about **humans writing code or AI writing code**. It is about humans and intelligent tools working together. AI can provide speed and suggestions, while programmers provide creativity, reasoning, context, verification, and responsibility.

For computer science students, the goal should not be to avoid AI but to learn how to **use it intelligently without losing the ability to think, design, and code independently**.

Student's Articles

HACKERS VS. AI: THE NEW CYBERSECURITY BATTLEFIELD

When Artificial Intelligence Becomes Both the Weapon and the Shield



Anisha Sinha

Roll. No. : 25015(Final Year)

In today's connected world, cyber security has become a continuous battle between attackers and defenders. From smartphones and banking systems to cloud platforms and critical infrastructure, digital technology is everywhere. At the centre of this changing battlefield is **Artificial Intelligence (AI)**—a technology that is transforming both cyber-attacks and cyber defence.



Student's Articles

The Rise of AI-Powered Cyber Threats

AI is enabling cybercriminals to automate tasks, analyse large amounts of information and create more convincing attacks.

Generative AI can be used to produce realistic phishing emails, fake websites and personalised social-engineering messages, making fraudulent communication harder to identify.

Another growing concern is **deep fake technology**, which can generate realistic images, videos and voices for impersonation, fraud and misinformation. As synthetic content becomes more convincing, verifying digital identity is becoming increasingly important.

AI: The New Cyber security Shield

The same technology can also strengthen cyber defence. AI-powered systems can monitor networks, identify unusual behaviour and help security teams respond to threats more quickly

AI is increasingly being applied to:

Threat Detection – Identifying suspicious network activity.

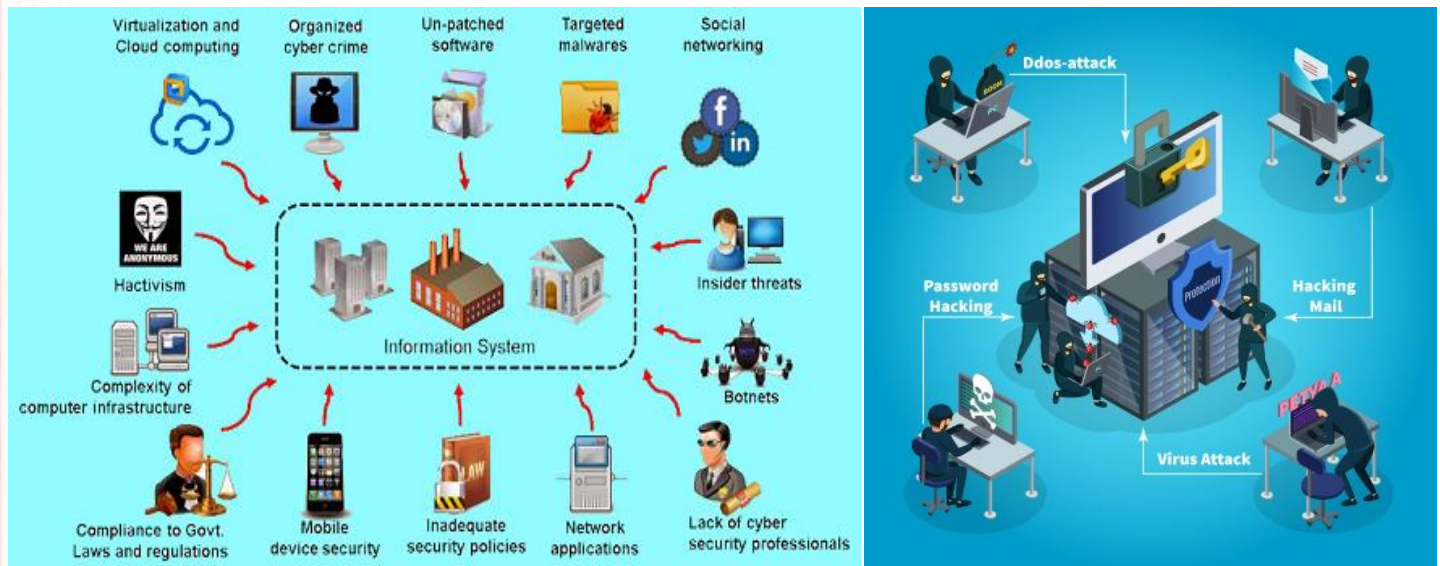
Malware Analysis – Detecting potentially harmful files.

Fraud Detection – Recognising unusual transactions and account activity.

Vulnerability Management – Identifying potential security weaknesses.

Student's Articles

Incident Response – Supporting faster investigation of security incidents.



DID YOU KNOW?

AI can analyse huge volumes of digital activity rapidly, helping security teams identify patterns that may indicate a cyber-threat.

The AI vs. AI Era

Cyber security is entering an era where **AI may operate on both sides of the battlefield**. Attackers can use AI for automation and deception, while defenders use it for detection, analysis and response.

ATTACKERS: AI → Automation → Deception → Exploitation

DEFENDERS: AI → Detection → Analysis → Response

This makes cyber security increasingly proactive, requiring organisations to continuously adapt to emerging threats.

Student's Articles

The Human Factor

Technology alone cannot guarantee security. Human behaviour remains a critical part of cyber security. Strong passwords, **multi-factor authentication**, **regular software updates**, **secure access controls** and **cyber security awareness** can significantly reduce risks.

AI is therefore more likely to **assist cyber security professionals rather than completely replace them**. Human expertise remains essential for investigation, ethical judgement and critical decision-making.



Student's Articles

Can AI Replace Cyber security Professionals?

AI can automate many repetitive and time-consuming security tasks, but cyber security professionals remain essential.

Security incidents often require contextual understanding, investigation, ethical judgement and strategic decision-making. Instead of completely replacing cyber security experts, AI is more likely to **change the way they work**.

Future cyber security professionals may increasingly work alongside AI systems, using them as intelligent assistants for monitoring, analysis and threat investigation while humans remain responsible for critical decisions.

The Future Cyber security Battlefield

The cyber security environment of the future will extend beyond traditional firewalls and antivirus software. Organisations will need a combination of **AI-powered detection, zero-trust security, strong identity management, encryption, continuous monitoring and cyber security education**.

Emerging technologies such as **post-quantum cryptography, autonomous security systems and AI-driven threat intelligence** could further transform digital defence.

The challenge will not simply be building more powerful AI. It will be ensuring that AI systems themselves are secure, transparent, responsibly deployed and capable of operating alongside human experts.

Student's Articles



KEY TAKEAWAYS

- ◆ AI is changing both cyber-attacks and cyber defence.
- ◆ Generative AI can make phishing and social engineering more convincing.
- ◆ AI can help security teams detect unusual behaviour and analyse threats.
- ◆ Human awareness remains an essential cyber security layer.
- ◆ The future will increasingly involve humans and AI working together.
- ◆ Cyber security must continuously evolve as attack techniques change.

Student's Articles



CONCLUSION

The battle between hackers and cyber security professionals is entering a new technological era. Artificial Intelligence has become both a powerful defensive shield and a potential offensive tool. Its impact will depend largely on how responsibly and effectively it is developed and used.

For Computer Science students, this changing landscape presents both a challenge and an opportunity. Tomorrow's cyber security professionals will need more than programming skills—they will need knowledge of **AI, networks, cloud computing, data privacy, ethical hacking and digital security**.

The future of cyber security will not be determined by AI alone. It will be shaped by the people who design it, secure it and use it responsibly.

“In the new cyber security battlefield, intelligence is the weapon—and responsibility is the defence.”

Student's Articles

AI at the Edge: Intelligence inside your Devices



Himanshu Yadav
Roll No.: 26073(3rd Year)

Artificial Intelligence is no longer limited to large cloud servers and data centres. Increasingly, AI capabilities are being embedded directly into smartphones, cameras, vehicles, industrial machines, wearables, and other connected devices. This approach is known as **Edge AI**, where data is processed and AI models are executed close to the location where the data is generated.

In a traditional cloud-based system, a device collects data and sends it to a remote server for processing. The result is then transmitted back to the device. Edge AI changes this architecture by moving at least part of the computation closer to the user or device.



This can make intelligent applications faster, more private, and less dependent on continuous internet connectivity.

Student's Articles

What Is Edge AI?

Edge AI combines **Artificial Intelligence with Edge Computing**. Edge computing moves computational resources closer to data sources, while AI enables devices to analyse that data and make intelligent decisions.

Consider a smart security camera. In a conventional cloud-based system, the camera may continuously transmit video to a remote server for analysis. With Edge AI, an AI model running on the camera or a nearby edge device can analyse the video locally and identify objects or unusual activity.

Only important information, such as an alert or selected image, may need to be sent to the cloud.

This reduces the amount of raw data that must travel across the network.

Why Move AI to the Edge?

1. Lower Latency: Applications such as autonomous systems, industrial robots, and driver-assistance technologies may require rapid responses. Processing data locally eliminates much of the delay associated with sending information to a remote server and waiting for a response.

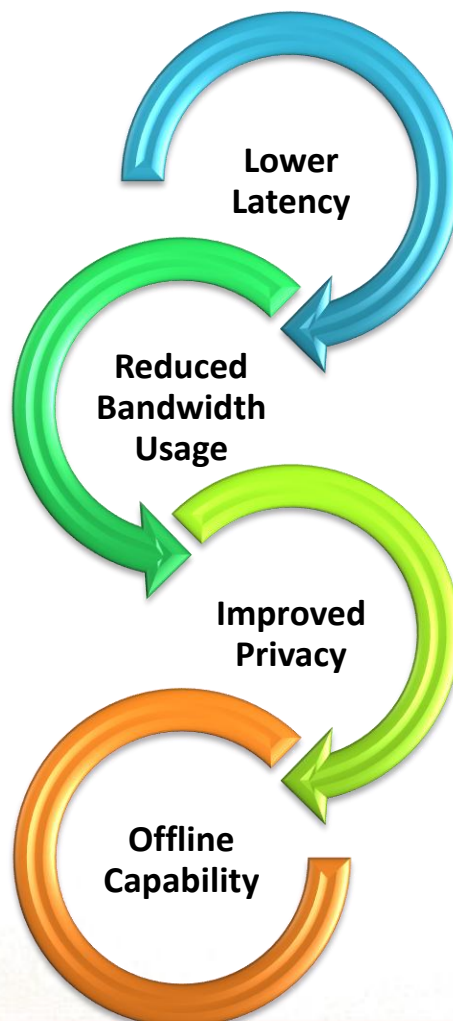
2. Reduced Bandwidth Usage: Video, audio, sensor readings, and other IoT data can generate enormous amounts of information. Local processing allows devices to analyse data before transmitting only relevant results, reducing network traffic.

Student's Articles

3. Improved Privacy: Sensitive information can sometimes remain on the device instead of being continuously uploaded to the cloud. For example, a smartphone can perform certain speech or image-processing tasks locally.

Local processing does not automatically guarantee privacy, but it can reduce the need to transmit raw personal data.

4. Offline Capability: Edge AI systems can continue performing certain tasks even when internet connectivity is unavailable or unreliable. This is particularly useful for remote industrial environments, vehicles, and field applications.



Student's Articles

How Does Edge AI Work?

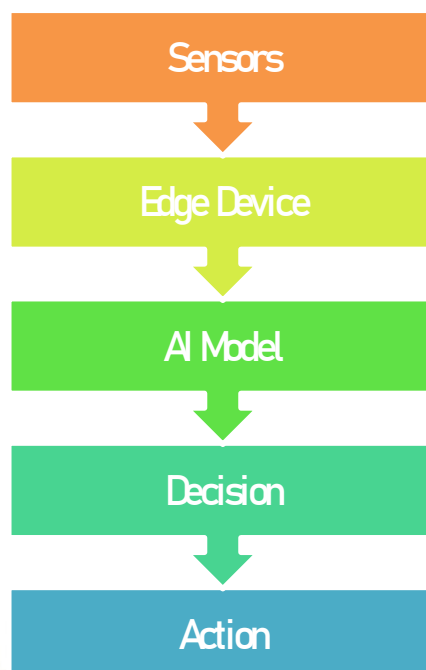
A typical Edge AI system consists of several components:

Sensors collect information such as images, temperature, motion, sound, or pressure. The edge device then processes this information using a trained AI model.

Because edge devices generally have fewer computational resources than large cloud servers, AI models often need to be optimised.

Techniques such as **model quantisation, pruning, knowledge distillation, and hardware acceleration** can reduce the memory and computational requirements of AI models.

For example, quantisation can represent model parameters using lower-precision numerical formats, potentially reducing memory consumption and improving inference speed.



Student's Articles

Edge AI and the Internet of Things

Edge AI has strong connections with the **Internet of Things (IoT)**. IoT devices generate large volumes of sensor data, while Edge AI provides the ability to analyse that data locally.

Consider a smart manufacturing facility. Sensors attached to machines can continuously monitor vibration, temperature, pressure, and energy consumption. An Edge AI model can identify patterns associated with abnormal machine behaviour and generate an alert before a major failure occurs. This concept is known as **predictive maintenance**.

Similar applications can be found in smart agriculture, healthcare devices, traffic monitoring, retail systems, and environmental monitoring.

Challenges of Edge AI

Despite its advantages, Edge AI presents several technical challenges.

Limited Computing Power: Edge devices usually have less processing capability and memory than cloud servers.

Energy Constraints: Battery-powered devices must perform AI computation while consuming as little power as possible.

Model Optimisation: Large AI models may need to be compressed or redesigned before they can run efficiently on edge hardware.

Security: An AI model deployed on a physical device may be exposed to tampering or other security threats.

Model Updates: Maintaining and updating AI models across thousands or millions of distributed devices can be complex. These challenges require collaboration between AI, computer architecture, embedded systems, networking, and cybersecurity.

Student's Articles

The Future of Edge Intelligence

As AI models become smaller and specialised hardware becomes more capable, intelligence is likely to move further into everyday devices. Smartphones, vehicles, cameras, wearables, robots, and industrial equipment can increasingly perform AI inference without depending entirely on remote servers.



For computer science students, Edge AI represents an intersection of several important areas: **Artificial Intelligence, Machine Learning, IoT, Embedded Systems, Computer Architecture, Networking, and Cybersecurity.**

In Conclusion, Edge AI represents a shift from “**send data to intelligence**” toward “**bring intelligence closer to the data.**” By processing information locally, devices can achieve lower latency, reduce bandwidth requirements, support certain offline operations, and potentially improve data privacy.

The future of intelligent computing will not exist entirely in the cloud or entirely on individual devices. Instead, increasingly sophisticated systems will distribute intelligence across **devices, edge infrastructure, and cloud platforms.**

Student's Articles

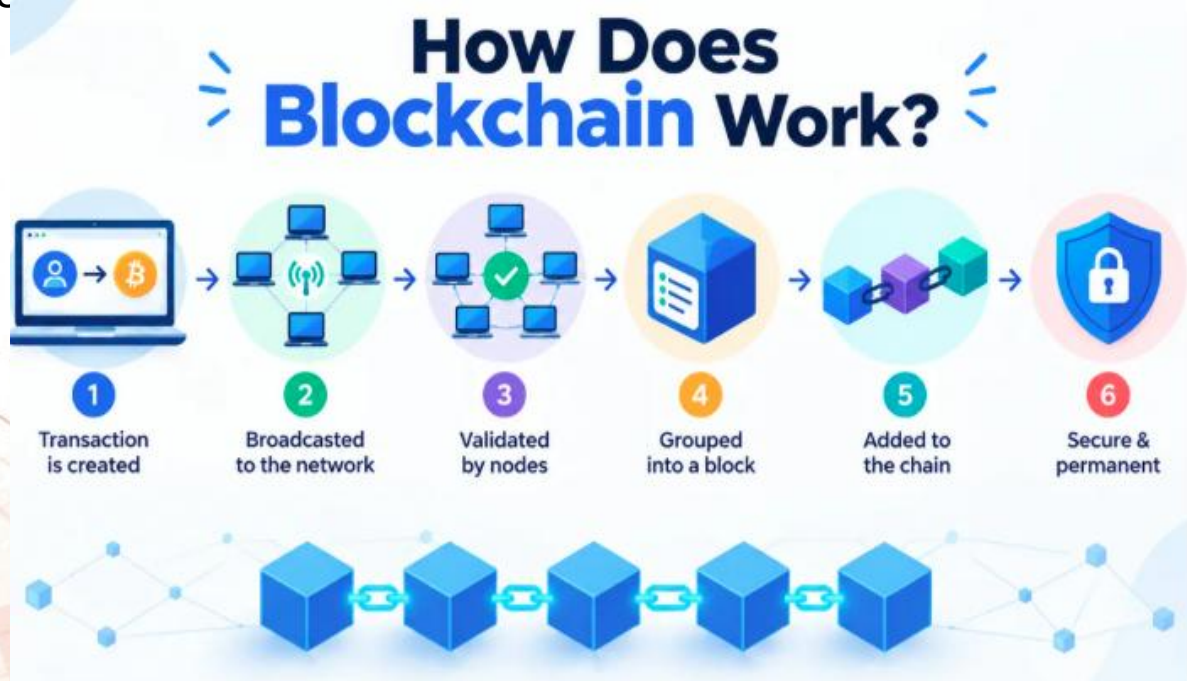
Blockchain beyond Bitcoin: Finding Practical Applications



Anjali Sharma
Roll No.: 25016(Final Year)

For many students, the word **blockchain** immediately brings Bitcoin and cryptocurrency to mind. However, blockchain is fundamentally a computer science concept involving **distributed systems, cryptography, databases, networking, and security**. Its applications extend far beyond digital currencies.

A blockchain can be understood as a distributed digital ledger in which records are grouped into blocks and cryptographically linked. Instead of relying entirely on one central database, multiple participating computers can maintain and verify the ledger.



Student's Articles

Consider a simple project where certificates are issued digitally. When a certificate is created, relevant information can be recorded as a transaction. Multiple transactions can be grouped into a block. Each block contains a cryptographic reference to the previous block, creating a chain.

A simplified flow for the working of a blockchain is as follows:



If someone attempts to modify an earlier record, the cryptographic relationship between blocks can reveal that the data has been altered.

1. Blockchain for Student Certificates

One of the practical applications relevant to colleges is **digital certificate verification**.

Suppose a student completes a course, internship, or certification. Instead of depending entirely on a physical certificate or manually contacting the institution for verification, the institution could issue a digitally verifiable credential.

A simplified system could work as:



Student's Articles

The blockchain could store a verification record or cryptographic hash rather than the student's complete personal information. A student could develop a prototype using a web application, database, blockchain platform, and a QR code that allows an authorised user to verify the certificate.

2. Blockchain and Supply Chain Tracking

Imagine ordering a laptop online. The product may pass through a manufacturer, distributor, warehouse, delivery company, and retailer before reaching the customer.

A blockchain-based system could record important events at each stage:



Each authorised participant can add relevant information to the shared record. This can make it easier to trace the history of a product.

3. Blockchain and Cybersecurity

Blockchain also demonstrates how cryptography can be used to protect data integrity.

Instead of storing information as ordinary records, systems can use **hash functions** to generate unique representations of data. If the original information changes, the resulting hash also changes. However, hashing does not mean that blockchain data is automatically private. Data visibility depends on the blockchain architecture and access controls.

Student's Articles

The blockchain could store a verification record or cryptographic hash rather than the student's complete personal information. A student could develop a prototype using a web application, database, blockchain platform, and a QR code that allows an authorised user to verify the certificate.

2. Blockchain and Supply Chain Tracking

Imagine ordering a laptop online. The product may pass through a manufacturer, distributor, warehouse, delivery company, and retailer before reaching the customer.

A blockchain-based system could record important events at each stage:

Each authorised participant can add relevant information to the shared record. This can make it easier to trace the history of a product.

3. Blockchain and Cybersecurity

Blockchain also demonstrates how cryptography can be used to protect data integrity.

Instead of storing information as ordinary records, systems can use **hash functions** to generate unique representations of data. If the original information changes, the resulting hash also changes. However, hashing does not mean that blockchain data is automatically private. Data visibility depends on the blockchain architecture and access controls.

Student's Articles

4. Smart Contracts: Programs That Execute Conditions

A particularly interesting concept for programmers is the **smart contract**. A smart contract is a program deployed on a blockchain that executes predefined rules when specified conditions are satisfied.

For example, consider an online internship platform:

Internship Completed → Supervisor Approval → Condition Verified → Certificate/Payment Process Initiated

Instead of manually processing every step, a smart contract could automate selected actions. However, smart contracts must be carefully tested because programming errors can lead to incorrect execution.

5. Blockchain + IoT

The combination of blockchain and the **Internet of Things (IoT)** creates another interesting area for student projects.

Imagine sensors monitoring the temperature of medicines during transportation. The IoT devices generate temperature readings, while a blockchain-based system can record important events in a tamper-evident manner. A simplified architecture could be represented as:



A student could develop a prototype where sensor data is collected, selected readings are recorded, and users can view the history through a web application. This project combines **IoT, networking, databases, cybersecurity, and blockchain**.

Student's Articles

4. Smart Contracts: Programs That Execute Conditions

A particularly interesting concept for programmers is the **smart contract**. A smart contract is a program deployed on a blockchain that executes predefined rules when specified conditions are satisfied.

For example, consider an online internship platform:

Internship Completed → Supervisor Approval → Condition Verified → Certificate/Payment Process Initiated

Instead of manually processing every step, a smart contract could automate selected actions. However, smart contracts must be carefully tested because programming errors can lead to incorrect execution.

5. Blockchain + IoT

The combination of blockchain and the **Internet of Things (IoT)** creates another interesting area for student projects.

Imagine sensors monitoring the temperature of medicines during transportation. The IoT devices generate temperature readings, while a blockchain-based system can record important events in a tamper-evident manner. A simplified architecture could be represented as:

A student could develop a prototype where sensor data is collected, selected readings are recorded, and users can view the history through a web application. This project combines **IoT, networking, databases, cybersecurity, and blockchain**.

Student's Articles

Is Blockchain Always Necessary?

The blockchain is **not a replacement for every database**. For example, a college has one trusted administrator who manages a student database. A conventional relational database may be simpler and more efficient. Blockchain becomes more interesting when:

Multiple parties need to share a common record.

Participants may not completely trust one another.

Data integrity and traceability are important.

An auditable history of transactions is required.

Programmable rules need to be executed through smart contracts.

Understanding this distinction is an important part of good system design.

Challenges for Developers

Building blockchain applications also introduces to challenges such as:

Scalability: Blockchain networks may process transactions differently from conventional databases.

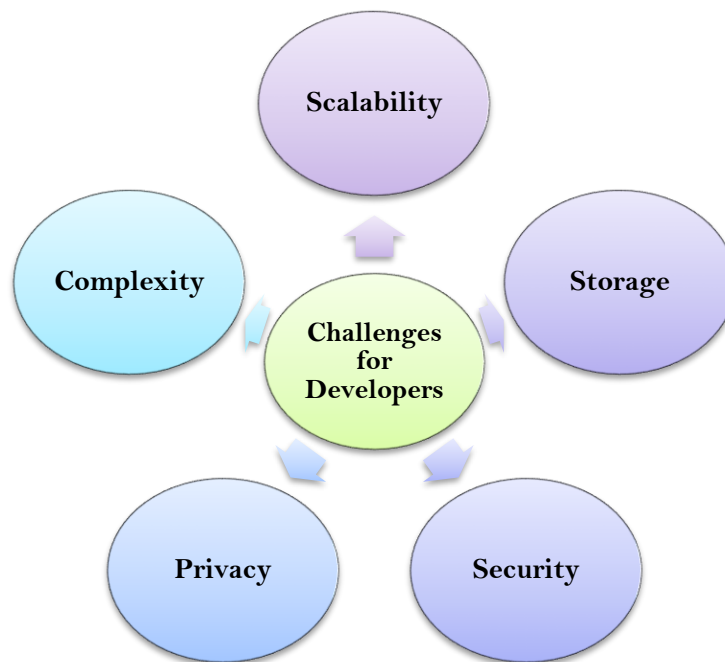
Storage: Storing large files directly on a blockchain is generally inefficient.

Security: Smart contracts and blockchain applications can contain programming vulnerabilities.

Privacy: Transparent records may not be suitable for sensitive information.

Complexity: Developers must understand cryptography, distributed systems, networking, and blockchain architecture.

Student's Articles



Therefore, learning blockchain requires more than simply learning how to create cryptocurrency transactions.

In conclusion, Blockchain is much more than Bitcoin. From certificate verification and supply-chain tracking to smart contracts and IoT security, its concepts can be applied to several real-world computing problems.

The real value of learning blockchain is not simply knowing how cryptocurrency works. It is understanding how **cryptography, distributed computing, databases, networking, and programming can work together to create systems where data can be shared and verified without relying entirely on a single central authority.**

The most important question for a future software developer is therefore not:

“Where can I use blockchain?”

but:

“What problem does blockchain solve better than a conventional system?”

Department's Achievements



DRONACHARYA
College of Engineering



Congratulations! for securing 1st Position

ShlokaDecode 2.0

under INNOVISION'25

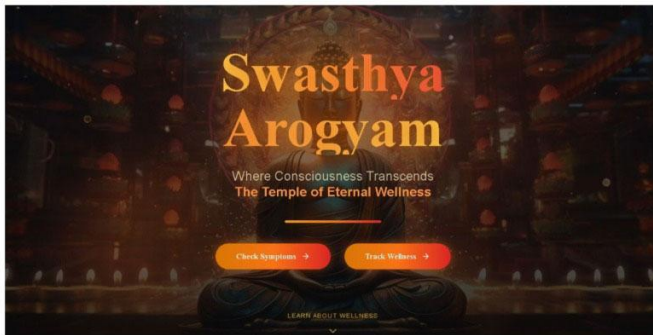
Organized by: NSUT, Delhi



Participants

Divik Arora , Roll No: 27058(CSE) 2nd year

Aarushi Prajapati, Roll No: 27002(CSE) 2nd year



INNOVISION'25

Divika Arora and Aarushi Prajapati (CSE 2nd Year) secured 1st Position at ShlokaDecode 2.0 under INNOVISION'25 Organized by: NSUT, Delhi.

Department's Achievements



B.Tech Second Year Students Presented Paper
THE 5TH INTERNATIONAL CONFERENCE ON
ELECTRICAL, COMPUTER AND ENERGY TECHNOLOGIES (ICECET-2025)
03-06 July, 2025 | PARIS/FRANCE (Virtual Mode)



Riya
Roll No : 26158 | CSE



Kanishka
Roll No : 26081 | CSE

Under The Guidance of
Dr. Monika Rani,
Assistant Professor (CSE)



For An Oral Presentation with Paper Titled: Advancements in Geographic Information System

ICECET-2025 Presentation

Riya and Kanishka(CSE Department)
presented “Advancements in Geographic
Information System” at ICECET-2025, Paris.

Department's Achievements





Congratulations!

Students Won The 1st PRIZE

Among 1800+ (Online Participants on Unstop) & Top 15 Teams In the (Offline Round) At

 **IIITD** - Indraprastha Institute of Information Technology Delhi 

Innovortex 3.0 (24 Hours Hackathon) held at 4th-5th October, 2025



Organized By:
TechNeeds IGDTUW

Project Name:
Inception (Dynamic NFTs & Omnichain Gaming Arena)



Team Members:

- DIVIK ARORA | Roll No : 27058
- HARSH DIXIT | Roll No : 27068
- HARDIK DHINGRA | Roll No : 27064
- AARUSHI PRAJAPATI | Roll No : 27002

Department of CSE
BATCH: 2024-2028

National-Level Hackathon Winners

CSE students won 1st Prize at the national-level Innovortex 3.0 hackathon, IIIT Delhi, for their project “Inception: Dynamic NFTs & Omnichain Gaming Arena.”