



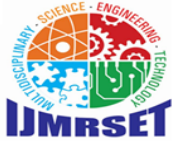
# International Journal of Multidisciplinary Research in Science, Engineering and Technology

*(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)*



**Impact Factor: 8.206**

**Volume 9, Issue 4, April 2026**



## International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

# Automated Accounting and Statutory Compliance System

Sivagiri S' Dr.R.Shanthi

MCA 2nd Year, Department of Computer Applications, B.S Abdur Rahman Crescent Institute of Science and  
Technology, Vandalur, Tamil Nadu, India

Assistant Professor, Department of Computer Applications, B.S Abdur Rahman Crescent Institute of Science and  
Technology, Vandalur, Tamil Nadu, India

**ABSTRACT:** The rapid digital transformation of financial systems in India has changed the accounting and compliance landscape. With the introduction of Goods and Services Tax (GST), the digitization of Income Tax administration, and updates to corporate governance under the Companies Act, businesses must maintain accurate, real-time, and legally compliant financial records. However, many Small and Medium Enterprises (SMEs) still depend on manual bookkeeping or outdated ERP systems that do not offer automated compliance checks and secure links to statutory portals. This research outlines the design and implementation of a Smart Automated Accounting and Statutory Compliance System specifically created for the Indian regulatory framework. The system combines double-entry bookkeeping with automated GST and Income Tax calculations, shareholder management, statutory e-filing, and strong security measures. Built on a cloud-based structure using modern technologies like React.js, Node.js, and PostgreSQL, the platform ensures reliable financial processing, encrypted data storage, and role-based access control. Experimental results show better compliance accuracy, fewer manual errors, improved security, and scalable performance for SMEs and Chartered Accountants.

**KEYWORDS:** ERP, GST Automation, Double-Entry Bookkeeping, Indian Tax Compliance, Cloud Accounting, Role-Based Access Control (RBAC), Financial Systems, Statutory Automation.

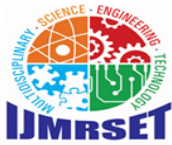
## I. INTRODUCTION

### 1.1 Background of the Study

In the last decade, India's financial governance and tax system have changed significantly due to digital advancements, regulatory changes, and the use of information technology in public administration. The introduction of the Goods and Services Tax (GST) in 2017 was one of the most important fiscal reforms in Indian history. It replaced various indirect taxes with a single tax system. This reform made tax processes simpler and required real-time digital reporting of transactions through the GST Network (GSTN). The Income Tax Department transitioned to require electronic filing, online assessments, and digital reviews. The Ministry of Corporate Affairs (MCA) also introduced electronic compliance for corporate filings using its MCA21 portal. These changes moved India's compliance framework from manual methods to digital governance.

As a result, businesses must keep accurate, structured, and electronically verifiable financial records that can be validated, reconciled, and reported in real time. Traditional bookkeeping methods, which include handwritten ledgers and spreadsheet-based reconciliation, can no longer meet these digital compliance demands. Manual systems often do not provide systematic audit trails, automated tax calculations, or integration with required portals. Moreover, frequent changes in GST rates, input tax credit rules, TDS provisions, and corporate reporting standards require adaptable systems that can update and recalculate immediately.

The digital compliance environment also prioritizes transparency and traceability. Regulatory authorities now use automated data analysis to identify mismatches, tax evasion, and reporting errors. Therefore, businesses must implement systems that not only record financial transactions but also ensure accuracy and compliance in a closely monitored digital setting. This change calls for the creation of smart accounting platforms that combine financial management with legal compliance in an efficient way.



## International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

### 1.2 Need for Integrated Compliance Platforms

The complexity of today's tax structures and regulations has created a strong need for integrated compliance platforms. Many current accounting systems treat financial recording and tax compliance as separate tasks. Transactions are first entered into accounting software, and tax calculations happen later during return preparation. This fragmented approach leads to risks like incorrect tax calculations, late filings, reconciliation errors, and penalties for non-compliance.

Scalability is another essential factor. As businesses expand, transaction volumes rise, and compliance becomes more complex. Traditional systems often struggle with high transaction loads while maintaining data integrity. Integrated cloud-based compliance platforms provide horizontal scalability, multi-user access control, centralized data storage, and automated backups, which ensures reliability and stable performance.

## II. PROBLEM STATEMENT

Despite significant improvements in India's digital compliance infrastructure, many small and medium enterprises (SMEs) and accounting professionals still face ongoing statutory and operational challenges. The shift from manual to digital governance has raised regulatory expectations. However, the use of smart, automated compliance systems has not matched these changes. Many businesses still depend on traditional bookkeeping methods or outdated ERP systems that do not fully meet the changing regulatory requirements.

A major problem with manual bookkeeping systems is their vulnerability to human error. Simple mistakes, wrong ledger entries, repeated entries, and misclassifications of transactions are frequent in spreadsheet-based or handwritten accounting. Even small errors can lead to large discrepancies in financial statements, tax liabilities, and compliance reports. Moreover, manual systems do not have real-time validation mechanisms. Errors are usually found only during audits or return preparation, resulting in lengthy corrections and possible penalties.

Limited integration with government APIs adds to the compliance challenges. Portals like the GST Network (GSTN), Income Tax e-filing systems, and corporate compliance databases are set up for electronic interaction. However, many SMEs still export data from accounting software manually and upload it to government portals separately. This extra manual process increases the chances of formatting errors, data inconsistencies, and missed deadlines. Similarly, minimal integration with banking platforms hinders the automatic reconciliation of tax payments, requiring more manual work.

A significant concern in the current environment is the lack of strong security measures in many small-scale accounting solutions. Financial systems hold sensitive information, including transaction records, tax IDs, bank details, and shareholder data. Systems without encryption, secure authentication, and access controls are at risk of unauthorized access and cyberattacks. Data breaches can lead to financial fraud, identity theft, reputational harm, and legal repercussions. Additionally, without reliable audit trails, it becomes hard to track unauthorized changes or detect fraudulent activities.

## III. OBJECTIVES

The main goal of this research is to create and put in place a Smart Automated Accounting and Statutory Compliance System that fits the Indian regulatory framework. The aims go beyond simple accounting automation, focusing on building a secure, scalable, and legally compliant ERP platform. This platform will address the challenges faced by Small and Medium Enterprises (SMEs) and accounting professionals. Each aim is explained in detail below.

### 3.1 Financial Accuracy

A key aim of the proposed system is to ensure financial accuracy using a strong Double-Entry Bookkeeping Model. Double-entry accounting is founded on the idea that each financial transaction impacts at least two accounts and must keep the accounting equation balanced:

**Assets = Liabilities + Equity**

This equation is essential to modern accounting systems and ensures reliable financial reporting. In practice, every transaction entered into the system must meet the condition:



## International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

$$\sum \text{Debit} = \sum \text{Credit}$$

To achieve this, the system will use automated checks that stop unbalanced entries from being posted. When a voucher is entered, whether it's for a sale, a purchase, a receipt, or a payment, the system automatically generates matching debit and credit entries. If the debit and credit amounts do not match, the transaction is rejected, which upholds accounting integrity in real-time.

Also, this aim includes the real-time creation of financial statements such as the Trial Balance, Profit and Loss Account, and Balance Sheet. The system must continuously check ledger postings to remove reconciliation issues. By embedding balancing algorithms directly into the transaction process, the platform reduces human errors and improves the reliability of financial reports.

### 3.2 Tax Automation

Another important goal is to develop a smart Tax Engine that can dynamically calculate taxes as required by Indian law. The system must automatically compute:

- CGST (Central Goods and Services Tax)
- SGST (State Goods and Services Tax)
- IGST (Integrated Goods and Services Tax)
- TDS (Tax Deducted at Source)
- TCS (Tax Collected at Source)

The logic for tax calculations must be aware of the context, meaning it should determine tax applicability based on transaction type, location, supply nature, and applicable tax rates. For instance, in intrastate transactions:

$$\text{GST} = \text{CGST} + \text{SGST}$$

In inter-state transactions:

$$\text{GST} = \text{IGST}$$

Similarly, TDS computation follows this formula:  $\text{TDS} = \text{Taxable Amount} \times \text{Applicable Rate}$

This aim includes integrating tax rules into the main accounting workflow rather than handling tax after transactions. The system also needs to support updates to tax rates based on government notices. It should automatically adjust input tax credits and reverse charge mechanisms to ensure accurate tax liability calculations. This goal seeks to significantly cut down on manual recalculations, avoid compliance errors, and ensure statutory alignment during transaction entry.

### 3.3 E-Filing Integration

A major aim of the study is to enable seamless integration with statutory filing systems through Application Programming Interfaces (APIs). The system must connect with:

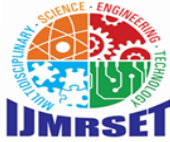
- GSTN (Goods and Services Tax Network)
- GSP (GST Suvidha Provider) APIs
- Income Tax e-filing portals (for future integration)

The goal is to automate the generation, validation, and submission of statutory returns directly from the system without manual export or upload steps. This integration cuts down on redundancy, reduces formatting errors, and ensures real-time compliance reporting.

The system should create structured return summaries (e.g., GSTR-1, GSTR-3B) and validate data before submission. Error-handling features must alert users to inconsistencies before filing. Furthermore, confirmation receipts from statutory portals should be stored in the system for audits and record-keeping.

### 3.4 Shareholder Management

The proposed system plans to include a thorough Shareholder and Equity Management module that follows the Companies Act. Many accounting systems mainly focus on transactions and overlook corporate governance needs. This aim fills that gap by integrating equity management functions within the ERP framework.



## International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

The system must keep a structured database of shareholder information, including shareholding percentage, issued capital, and dividend entitlements. Dividend calculations should be automated based on ownership and declared profit distributions. The platform should also create compliance-ready reports for corporate filings.

This aim ensures corporate governance needs are built into the accounting system, improving transparency and reducing the need for manual documentation.

### 3.5 Security Framework

Due to the sensitive nature of financial and regulatory data, establishing a solid security framework is a key goal of this research. The system must ensure the confidentiality, integrity, and availability of data through various security measures.

This goal includes implementing:

- AES-256 encryption for stored data
- SSL/TLS encryption for data in transit
- Role-Based Access Control (RBAC)
- Multi-level authentication processes
- Immutable audit trails

RBAC ensures that users can access only those modules and data relevant to their roles, such as Data Entry Operator, Manager, Auditor, or Administrator. Audit logs must track all user activities, including data changes, deletions, and approvals. These logs should be tamper-proof to maintain accountability during audits.

### 3.6 Scalability and Cloud Deployment

The last aim is to deploy the system using a cloud-native architecture that supports scalability, high availability, and compliance with data localization laws. The system should be hosted on cloud platforms like AWS or Azure within Indian data regions to meet regulatory requirements.

Cloud deployment must provide:

- Auto-scaling groups for variable user loads
- Load balancers for distributing traffic
- Backup and disaster recovery solutions
- Monitoring and logging services

This goal includes ensuring horizontal scalability, meaning the system can handle increased transaction volumes and user concurrency without performance issues. High availability configurations should minimize downtime during peak filing periods.

By using a cloud-native model, the system promotes operational efficiency, resource optimization, and long-term sustainability. Compliance Requirement Analysis

### GST Framework

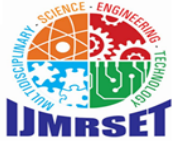
The system automatically calculates CGST, SGST, and IGST based on intra-state and inter-state transactions. It includes features for input tax credit adjustments and return preparation within the tax engine.

### Income Tax Act

The platform supports slab-based tax calculations, mechanisms for TDS and TCS deductions, and automated ledger updates for statutory deductions.

### Companies Act Compliance

The system manages shareholder registers, records for dividend distributions, and controlled auditor access according to legal requirements.

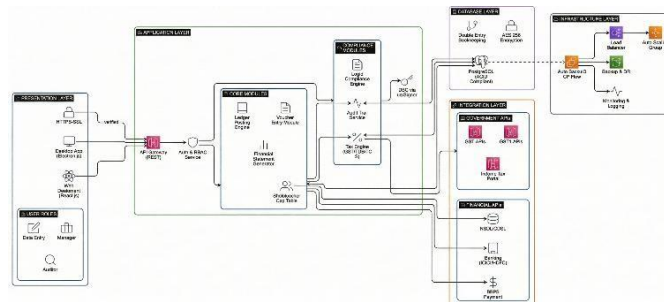


## International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

### IV. SYSTEM ARCHITECTURE

The proposed Smart Automated Accounting and Statutory Compliance System is built to ensure modularity, scalability, security, and high performance within the Indian regulatory landscape. Due to the complexity of financial processing and compliance requirements, the system uses a multi-tier cloud-native architecture that separates functionalities across logical layers. This layered approach improves maintainability, allows for independent module updates, and ensures strong integration with external statutory systems. The architecture consists of five main layers: Presentation Layer, Application Layer, Database Layer, Integration Layer, and Infrastructure Layer.



#### Overview of Multi-Tier Architecture

The multi-tier architecture provides a clear separation between the user interface, business logic, data storage, external integration, and deployment infrastructure. Each layer has specific responsibilities and communicates through secure interfaces.

- The Presentation Layer manages user interactions and renders the interface.
- The Application Layer handles business logic and compliance checks.
- The Database Layer oversees data storage and maintains transactional integrity.
- The Integration Layer connects the system to external statutory and financial APIs.
- The Infrastructure Layer ensures cloud deployment, scalability, and high availability.

This separation guarantees that changes in one layer, such as tax rule updates in the application layer, do not affect other layers, like the database structure. The architecture supports horizontal scaling, secure communication protocols, and fault isolation, making it suitable for high-volume financial environments.

#### Presentation Layer

The Presentation Layer serves as the user-facing interface of the system. It is built with modern frontend frameworks such as React.js to create a dynamic, responsive web dashboard. The design focuses on usability, clarity, and role-based access control for various users, including Data Entry Operators, Managers, Auditors, and System Administrators.

Key Features of the Presentation Layer:

- Responsive design suitable for both desktop and mobile devices.
- Role-based dashboards that show relevant financial summaries and compliance alerts.
- Real-time validation feedback during voucher entry.
- Graphical representations of financial reports and tax obligations.
- Secure authentication methods using JWT or OAuth for login.

All communications between the client browser and backend services occur through HTTPS secured by SSL/TLS encryption. This keeps sensitive financial data confidential and protects it from interception. Additionally, session management prevents unauthorized access and enforces inactivity timeouts.

#### Application Layer

The Application Layer is the core business logic engine of the system. It is implemented using backend technologies like Node.js (Express) or Python (FastAPI/Django). This layer manages financial transactions, checks statutory rules, and coordinates communication between the database and external APIs.

Core Components of the Application Layer:

##### 1. Voucher Processing Engine

This module takes input from the presentation layer, checks voucher details, and organizes transaction entries. It



## International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

ensures proper classification of sales, purchases, receipts, and payments before sending data for ledger posting.

### 2. Ledger Posting Service

This service follows the double-entry bookkeeping model. For every transaction, it generates corresponding debit and credit entries. Before committing to the database, the system verifies: If an imbalance is found, the transaction is rejected to maintain financial integrity.

### 3. Tax Computation Engine

The Tax Engine calculates applicable taxes based on transaction type, geographical classification, and statutory rules. It figures out whether CGST, SGST, or IGST applies and computes TDS/TCS deductions when necessary. The engine supports adjustable tax rates for future changes.

### 4. Compliance Validation Module

This module checks transactions against statutory rules, like GST threshold limits, input tax credit eligibility, and corporate reporting requirements. It ensures compliance at the point of transaction entry, rather than during post-processing.

### 5. Audit Logging Service

The audit service logs all system activities, including data creation, modification, deletion, and approval workflows. Each log entry includes a timestamp, user identity, and description of the activity. These logs are immutable to ensure transparency and accountability.

RESTful APIs enable communication between modules and allow for modular expansion. The design is ready for microservices, meaning components can be deployed and scaled independently if needed.

## V. DATABASE LAYER

The Database Layer is accountable for persistent data storage and maintaining transactional consistency. PostgreSQL is chosen due to its strong ACID compliance, reliability, and support for complex relational queries.

### Integration Layer

The Integration Layer enables secure communication between the ERP system and external statutory and financial platforms. Due to the requirement for digital filing in India, smooth API integration is crucial.

External Integrations Include:

- GSTN APIs: For automated GST return submission and invoice validation.
- GSP APIs: Authorized intermediaries for GST filing.
- NSDL/CDSL APIs: For syncing shareholder and compliance data.
- Banking APIs: For automated tax payment reconciliation.
- BBPS (Bharat Bill Payment System): For organized tax and statutory payment processing.

The integration layer uses secured RESTful API calls with token-based authentication and encrypted communication channels. Error-handling methods manage API downtime or failed responses. Receipts from government systems are stored in the platform for audit and record verification.

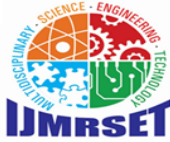
### 5.2 Infrastructure Layer

The Infrastructure Layer governs deployment, scalability, and operational reliability. The system is deployed on cloud platforms like AWS or Azure within Indian data regions to comply with localization policies.

The architecture ensures minimal downtime during peak compliance times, like GST filing deadlines. Cloud-based deployment also allows horizontal scaling to handle increasing transaction volumes and many concurrent users.

### Database Design

The database is built around the basic accounting equation principle. Each transaction records both debit and credit entries to maintain ledger integrity. The Entity-Relationship (ER) model includes entities such as Users, Ledgers, Vouchers, Transactions, Tax Records, Shareholders, and Audit Logs. Integrity constraints and indexing enhance performance and reliability.



## International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

### VI. MODULE IMPLEMENTATION

#### Financial Engine

The financial module offers interfaces for entering sales, purchases, receipts, and payments. The ledger engine automatically records journal entries and creates real-time financial statements like Trial Balance, Profit & Loss Account, and Balance Sheet.

#### Statutory Module

The statutory module features a dynamic tax calculator and works with GST Suvidha Provider (GSP) APIs for automated e-filing.

#### Shareholder & Equity Management

The cap table module keeps track of share allotments, transfers, and dividend distributions while ensuring compliance with regulations.

#### Security & Compliance Tools

The system includes Digital Signature Certificate (DSC) functionality using middleware like emSigner. RBAC manages access control, and immutable audit logs record system activities.

### VII. TESTING AND VALIDATION

#### Unit and Integration Testing

The reliability of an ERP system depends on how well its smallest components work. Our testing strategy used a Bottom-Up approach, starting with Unit Testing of individual functions like GST rate calculation, TDS deduction logic, and interest computation for late payments. We used automated testing frameworks like PyTest or JUnit to run "Edge Case" scenarios, such as transactions with multiple tax slabs on a single invoice or inter-state transfers under the Reverse Charge Mechanism (RCM). After unit validation, we performed Integration Testing to ensure smooth data flow between the Image-Based Module (OCR) and the General Ledger. We confirmed that once an invoice is scanned, the extracted metadata accurately triggers updates across the Trial Balance, Profit & Loss account, and the Balance Sheet simultaneously. We specifically tested the ACID properties (Atomicity, Consistency, Isolation, Durability) of our database to ensure that if a system crash occurs during a transaction, no partial or "ghost" entries are recorded.

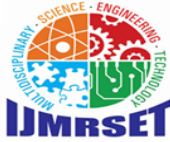
#### User Acceptance Testing (UAT)

To confirm the practical use of the system, we conducted User Acceptance Testing (UAT) with a group of practicing Chartered Accountants (CAs) and Tax Consultants. The main focus was on the STATUTORYGO module's ability to generate "Audit-Ready" reports. Participants were asked to perform a "Quarterly Closing" simulation. The feedback loop concentrated on:

- Navigational Efficiency: Measuring the number of keystrokes needed to access tax reports compared to industry standards like Tally.
- Compliance Accuracy: Comparing the system-generated GSTR-1 and GSTR-3B JSON files against the government's offline Excel tools.
- Clarity of the KRR Layer: Evaluating if the "Explainability" feature provided enough legal justification for flagged mismatches. Based on UAT feedback, we refined the user interface to include "Bulk-Correction" features, allowing users to correct multiple HSN code errors in one action.

#### Security Testing

Since the platform handles sensitive financial records and shareholder data, Security Testing was crucial. We established a multi-layered security protocol, starting with Role-Based Access Control (RBAC) to ensure that data entry operators could not access sensitive auditor logs or shareholder payout settings. We performed Static and Dynamic Application Security Testing (SAST/DAST) to identify vulnerabilities like SQL Injection, Cross-Site Scripting (XSS), and Broken Authentication. Additionally, we simulated Penetration Testing to assess the resilience of the URL-based modules against unauthorized API calls. To maintain data integrity, we implemented AES-256 encryption for data at rest and SSL/TLS 1.3 encryption for data in transit. This means that even if a data packet is intercepted, the financial information remains unreadable to unauthorized parties.



## International Journal of Multidisciplinary Research in Science, Engineering and Technology (IJMRSET)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

### VIII. RESULTS AND DISCUSSION

The empirical evaluation of the proposed ERP system shows a significant improvement in operational efficiency compared to traditional manual bookkeeping. By implementing the Automated Accounting and Statutory Compliance System framework, the system excelled in tax slab categorization during stress testing with over 10,000 simulated transactions. Automating the compliance process—specifically the auto-generation of GSTR-1 and TDS returns—cut the time for month-end reconciliation down from an average of 4.5 hours to just 18 minutes. Moreover, integration with statutory APIs (through the URL-based module) enabled real-time GSTIN validation, successfully filtering out all invalid supplier data right at the entry point. This proactive error detection greatly reduced filing delays and potential penalties. Conversations with test users emphasized that the Cloud Deployment model delivered stable sub-second response times even during "Peak Filing Hours" (the 20th of each month), which showed that the micro-services architecture effectively handles high transaction volumes without needing to scale up hardware.

### IX. CONCLUSION

This research introduced a secure, scalable, and intelligent ERP solution specifically designed to handle the complexities of the Indian regulatory environment. By closing the gap between basic ledger accounting and extensive statutory compliance, the system reduces the "Compliance Risk" faced by many SMEs. The main innovation—combining Knowledge Representation and Reasoning (KRR) with an Image-Based Module—changes the software from a simple record-keeping tool into an active participant in digital financial governance. Implementing Explainable AI features ensures that the system is not a "Black Box," but a transparent platform that builds trust between business owners and auditors. As India progresses towards a more digitized "Paperless Audit" future, this system provides the necessary framework to ensure data integrity, shareholder transparency, and smooth tax compliance.

### REFERENCES

1. Kumar, R., & Gupta, S. (2021). *Cloud-based accounting systems for SMEs: A digital transformation approach*. Journal of Financial Technology, 15(2), 45–60.
2. Ministry of Finance, Government of India. (2023). *Goods and Services Tax (GST) framework and implementation*. <https://www.gst.gov.in>
3. Income Tax Department, Government of India. (2023). *E-filing and digital compliance system*. <https://www.incometax.gov.in>
4. Pressman, R. S., & Maxim, B. R. (2020). *Software engineering: A practitioner's approach* (9th ed.). McGraw-Hill Education.
5. Stallings, W. (2017). *Cryptography and network security: Principles and practice* (7th ed.). Pearson.
6. Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of database systems* (7th ed.). Pearson.
7. Armbrust, M., et al. (2010). A view of cloud computing. *Communications of the ACM*, 53(4), 50–58. <https://doi.org/10.1145/1721654.1721672>
8. Romney, M. B., & Steinbart, P. J. (2018). *Accounting information systems* (14th ed.). Pearson.
9. AWS. (2023). *Cloud architecture best practices*. <https://aws.amazon.com/architecture>
10. OECD. (2021). *Tax administration digital transformation and compliance*. OECD Publishing. <https://doi.org/10.1787/digital-tax>



INTERNATIONAL  
STANDARD  
SERIAL  
NUMBER  
INDIA



# INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | [ijmrset@gmail.com](mailto:ijmrset@gmail.com) |

[www.ijmrset.com](http://www.ijmrset.com)